NASA Technical Memorandum 104557

p. 329

PC-SEAPAK User's Guide

Version 4.0

Charles R. McClain, Gary Fu, Michael Darzi, and James K. Firestone

JANUARY 1992

National Aeronautics and
Space Administration

Goddard Space Flight Center
Greenbelt, MD

NASA Technical Memorandum 104557

PC-SEAPAK User's Guide

Version 4.0

Charles R. McClain
Oceans and Ice Branch
Laboratory for Hydrospheric Processes
Goddard Space Flight Center
Greenbelt, Maryland

Gary Fu
Michael Darzi
James K. Firestone
General Sciences Corporation
Laurel, Maryland

National Aeronautics and
Space Administration

Goddard Space Flight Center
Greenbelt, MD

1992

TABLE OF CONTENTS

INTRODUCTION

PC-SEAPAK is a user-interactive satellite data analysis package that is being developed at NASA/Goddard Space Flight Center (GSFC) and is designed to operate on an IBM PC-AT class microcomputer. Its primary applications are for the processing and interpretation of satellite imagery data from the Nimbus-7 Coastal Zone Color Scanner (CZCS) and the TIROS-N/NOAA satellite series' Advanced Very High Resolution Radiometer (AVHRR). It is a subset of the SEAPAK analysis system (Darzi et al., 1989; McClain et al., 1989 and 1991; Fuh et al, 1990; Firestone et al, 1990 and 1991) that runs on a DEC VAX-based system at the Oceans Computing Facility at GSFC. PC-SEAPAK is designed to provide the user with a user-friendly standardized interface that provides flexibility and allows the user to truly work with the data. The user interface is a modified version of an interface developed by General Sciences Corporation under another federal government contract. SEAPAK development activity is being supported by the Ocean Processes Branch at NASA/Headquarters.

The system described in this document is the fourth version of PC-SEAPAK. The original design of the system was completed in November, 1987 (Firestone and Chen, 1987; Firestone et al., 1989). The actual software development began in March, 1988. Certainly, there will be enhancements to the hardware configuration and to the software in the future. The structure of this manual is designed to be updated periodically as new versions are released, so sections are numbered independently of other sections and the manual is in a loose-leaf form. The manual includes sections which describe in some detail the various hardware and software components of the system, some discussions on particular data processing scenarios, an explanation of the PC-SEAPAK menu and the programs it contains, a reference section containing detailed descriptions of all the SEAPAK programs, references and a glossary.

The CZCS was the first spaceborne sensor designed specifically to measure the concentration of photosynthetic pigments and their degradation products in the ocean. It had six co-registered bands (5 visible and one in the thermal IR) with a swath and resolution (2200 km and 825 meters at nadir, respectively) similar to the NOAA AVHRR. The CZCS IR band did not work after about the first year and was only useful for thermal feature delineation when it did work. The derived products generated by the CZCS level-2 programs are upwelled water radiance at 443, 520 and 550 nm, aerosol radiance at 670 nm, pigment concentration, diffuse attenuation at 490 nm and Rayleigh radiance at 443 nm. A variety of programs have been developed which allow the user to derive and evaluate the input parameters required by the level-2 generation programs.

Since sea surface temperature (SST) is an important oceanographic parameter, a capability for handling SST fields from the AVHRR was developed. Many of the analysis tools developed for the CZCS derived product fields are also useful in analyzing SST

fields. The algorithm for calculating the SST values from the AVHRR brightness temperatures varies depending on the satellite (TIROS-N or NOAA-6, 7, 8, 9, 10, 11, 12). PC-SEAPAK supports most of the SST algorithms that have been published for different satellites. In addition, PC-SEAPAK also allows the user to enter coefficients for a generalized SST equation.

PC-SEAPAK is organized into several categories of programs in menus that include level-1 data ingest, CZCS level-2 analyses, statistical analyses, data extraction, remapping to standard projections, graphics manipulation, image board memory manipulation and general utilities. Most programs allow user interaction not only through the menu and command modes, but also allow the user to work within a program by using the mouse cursor to define pixels or areas of interest and the function keys from which subprocesses may be executed in any order and any number of times without exiting the main program. Most programs provide for ASCII file generation for further analysis in spreadsheets, graphics packages, etc.

Obtaining PC-SEAPAK and User Support: PC-SEAPAK is available from NASA's Computer Software Management and Information Center (COSMIC) at the Univ. of Georgia (382 East Broad St., Athens, GA 30601; tel., 404-542-3265). The PC-SEAPAK development activity is supported with funding from NASA Headquarters. As long as the program is funded, assistance to the user community will be provided. This support includes assistance with the hardware and software configuration and for the correction of PC-SEAPAK software errors. Recommendations regarding modifications of existing programs and development of additional programs will be considered on a case by case basis. If you need assistance, please contact either Charles McClain (tel., 301-286-5377; OMNET, C.MCCLAIN; SPAN, URCHIN::MCCLAIN) or Gary Fu (tel., 301-286-7107; SPAN, URCHIN::GFU).

Obtaining CZCS Data: The following types of CZCS data may be obtained from the CZCS archive at NASA/GSFC (Feldman et al., 1989):

Level 1 Full resolution, swath projection (unmapped), calibrated radiance data for all six CZCS bands in a single scene.

Level 1b Subsampled (every fourth pixel and line) level-1 data for bands 1 to 5; about 4km resolution.

Level 2 Derived geophysical parameters for a single, unmapped CZCS scene at 4km resolution.

Level 3 Level-2 composited, Earth-gridded (binned) data.

Format options for these products are summarized in Table 1.

Table 2 identifies the proper PC-SEAPAK programs to use for the various formats and media of the data products. The Cipher M990 tape drive with the Flagstaff Engineering tape utilities may be used for 9-track tapes; the Summus drive with its Gigasafe utilities may be used for the 8mm tapes. **Note that these Summus utilities do not handle "foreign" tapes.** (Summus is no longer in business, although many PC-SEAPAK users may own this drive.) For the M990, we recommend that 1,600 bit/inch tapes be requested since

Table 1. Data formats available for 9-track and 8mm tapes.

Format	Data	Procedure to Create Tape under VAX/VMS
CRT	Level 1	Special program
VAX backup	Levels 1,1b,2,3	BACKUP command
Archive foreign	Levels 1,1b,2,3	MOUNT/FOREIGN and COPY
Archive labeled	Levels 1,1b,2,3	MOUNT/LABEL & COPY (ANSI)

problems with reading tapes of higher densities have been reported.

In order to evaluate CZCS coverage, a browse capability has been developed which includes (1) software that allows the user to query a data base using a number of parameters including latitude and longitude ranges and time interval to determine the scenes available which satisfy the query criteria and (2) a set of 3 Panasonic video disks which contain all the CZCS pigment images. The browse software can be used with or without the Panasonic player and versions are available for a variety of machines including PC-AT compatibles. The advantage of using the Panasonic video player is that each image that satisfies the query is displayed and the user has the option of ordering the displayed scene before progressing to the next scene.

Orders to the archive can be initiated from within the browse session provided the system is on a SPAN node. Data requests may also be filed through OMNET. Comments regarding format specification and requirements should be included with any request. For information about the CZCS archive, contact Gene Feldman, Code 936, NASA/GSFC, Greenbelt, MD 20771 (tel., 301-286-9428; OMNET, G.FELDMAN; SPAN, MANONO::GENE). The browse package is available at no charge.

Obtaining AVHRR Data: To obtain NOAA AVHRR level-1b data, contact Will Gould, Room 100, NOAA/Satellite Data Service Division, 5627 Allentown Road, Camp Spring, MD 20746 (tel., 301-763-8400). Also, one should refer to Brown et al. (1985), Kidwell (1988), Planet (1988), and Weinreb et al. (1990) regarding AVHRR data formats and calibration.

Table 2. PC-SEAPAK programs for ingesting CZCS data products.

Program	Level	Format	Tape Media	Input
TPCZCS,TP2DSK	1	CRT	9-track	tape
DKCZCS,DSK2DSK	1	CRT	9-track	disk
DKCZCS,DSK2DSK	1	ANSI,foreign	9-track,8mm	disk
DSPIMG	1b,2	ANSI,foreign	9-track,8mm	disk
PSTIMG	3	ANSI,foreign	9-track,8mm	disk

1. SYSTEM CONFIGURATION

The updated configuration for the original PC-SEAPAK development is shown in Figure 1. (The availability, price, and performance of hardware components are subject to frequent changes. The user should contact vendors directly to obtain the most current product information. See Appendix for recommended hardware and vendor information.) The main computational engine is a COMPAQ 386 (20, 25, or 30 MHz) which uses the Intel 80386 microprocessor. An Intel 80387 and a Weitek floating-point coprocessors, a 60 megabyte hard disk, and two floppy disk drives (a 5.25 inch with 1.2 MB capacity and a 3.5 inch with 1.44 MB capacity) are included in the system. The 80387 coprocessor is necessary in this system since it can greatly increase the speed of floating-point calculations and all the PC-SEAPAK software is built around it. The Weitek coprocessor is 2 to 3 times faster than the 80387. It is an option to the PC-SEAPAK configuration and only a few programs in PC-SEAPAK are available to run with this coprocessor. The hard disk normally is used for storing files of PC-SEAPAK and files associated with any additional software development. The floppy drives are used for transferring files to and from other sources.

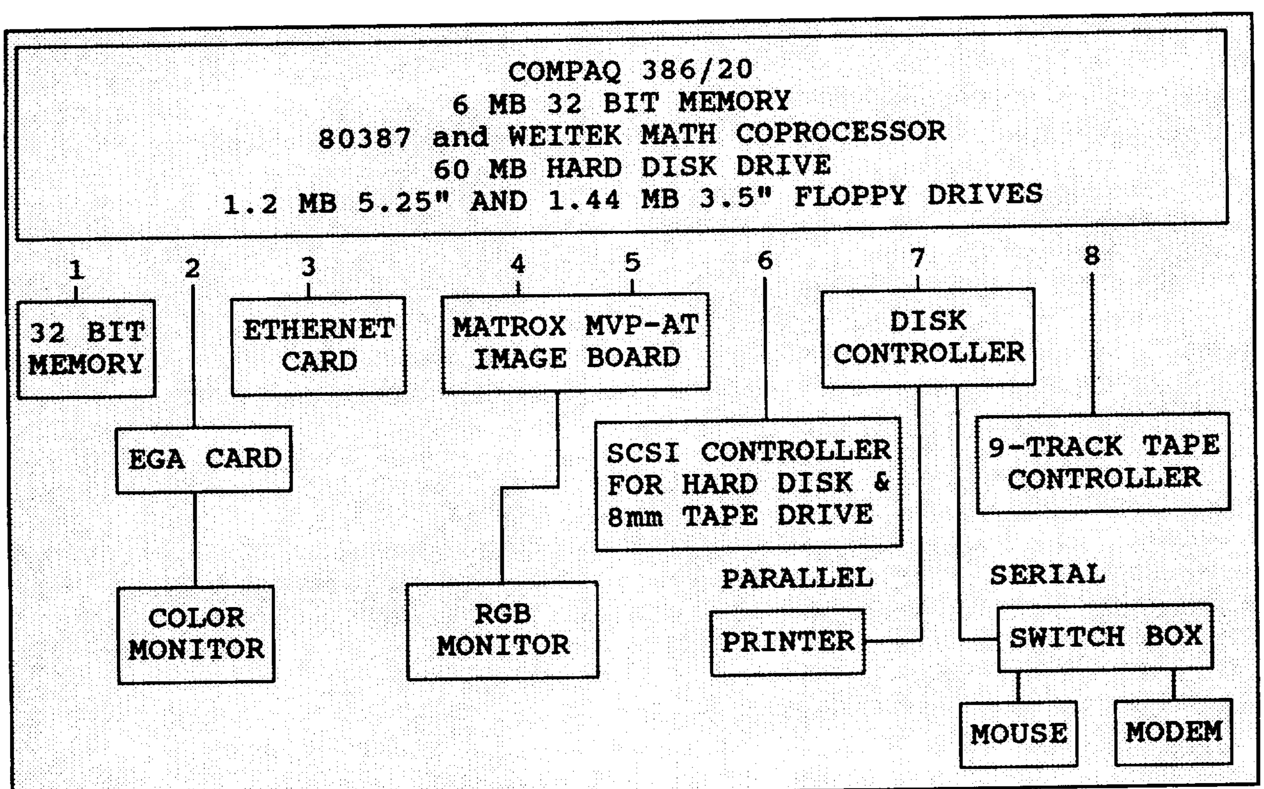

Figure 1. System configuration 1.

There are eight slots available for industry-standard expansion boards: one 32-bit memory slot, five full-sized 8-/16-bit slots, one full-sized 8-bit slot, and one half-size 8-bit slot. Currently, all the slots are used in this configuration.

In slot 1, on the 32-bit memory board, there are 6 MB of memory installed. Up to 16 MB memory may be put on the memory board. The extra memory (greater than the DOS 640 KB range) may be used as a virtual disk to improve the I/O throughput or may be used to run multi-tasking under expanded memory management applications like Quarterdeck's DESQview and QEMM. Also, it can be used for application programs that have been developed using Phar Lap's DOS-Extender and MicroWay's NDP Fortran-386 compiler to directly access the memory up to the limit.

In slot 2, the AST-3G EGA controller card is installed. A serial and a parallel port are supported by this card. However, this card requires another slot for these two ports and, since none are available, the two ports are not used in this configuration.

In slot 3, the 3Com EtherLink board 3C501 is installed. This board, combined with the DECnet-DOS network software, allows the workstation to be directly linked to the VAX cluster of the NASA/GSFC Oceans Computing Facility through a thin-wire Ethernet.

In slots 4 and 5, the Matrox MVP-AT image board is installed. This is a two-board image processing set with one megabyte of memory which is configured as four 512x512x8-bit frame buffers under PC-SEAPAK.

In slot 6, the Summus SCSI controller card is installed. The SCSI interface allows up to seven device drives to be daisy-chained into the same port. Summus supports a number of Imprimis WREN disk drives (capacities ranging from 100 MB to 1 GB), an 8mm tape drive and an erasable optical disk, all of which can be daisy-chained on the same SCSI interface. This is an excellent solution for unlimited storage, data backup and data exchange.

In slot 7, the Compaq's Multipurpose Fixed Disk Drive Controller card is installed. This card is included with the COMPAQ 386/20 system. It controls the 60 MB hard disk drive and the 5.25 and the 3.5 inches floppy diskettes drives. A parallel and a serial port are included in this card. In this configuration, the parallel port is connected to the printer and the serial port is used to connect to the modem and the mouse input device via a switch box.

In slot 8, the Flagstaff Engineering Tape Controller card is installed. This is used to control the 9-track M990 tape drive from Cipher Data Products, Inc.

Two new systems, the configuration of which is shown in Figure 2, have also been installed for running PC-SEAPAK at the NASA/GSFC Oceans Computing Facility. The new configuration is similar to that shown in Figure 1, except for the upgraded CPU and floating-point coprocessors, the built-in serial, parallel, and mouse ports, the VGA instead of the EGA controller, and the lack of a 9-track tape drive.

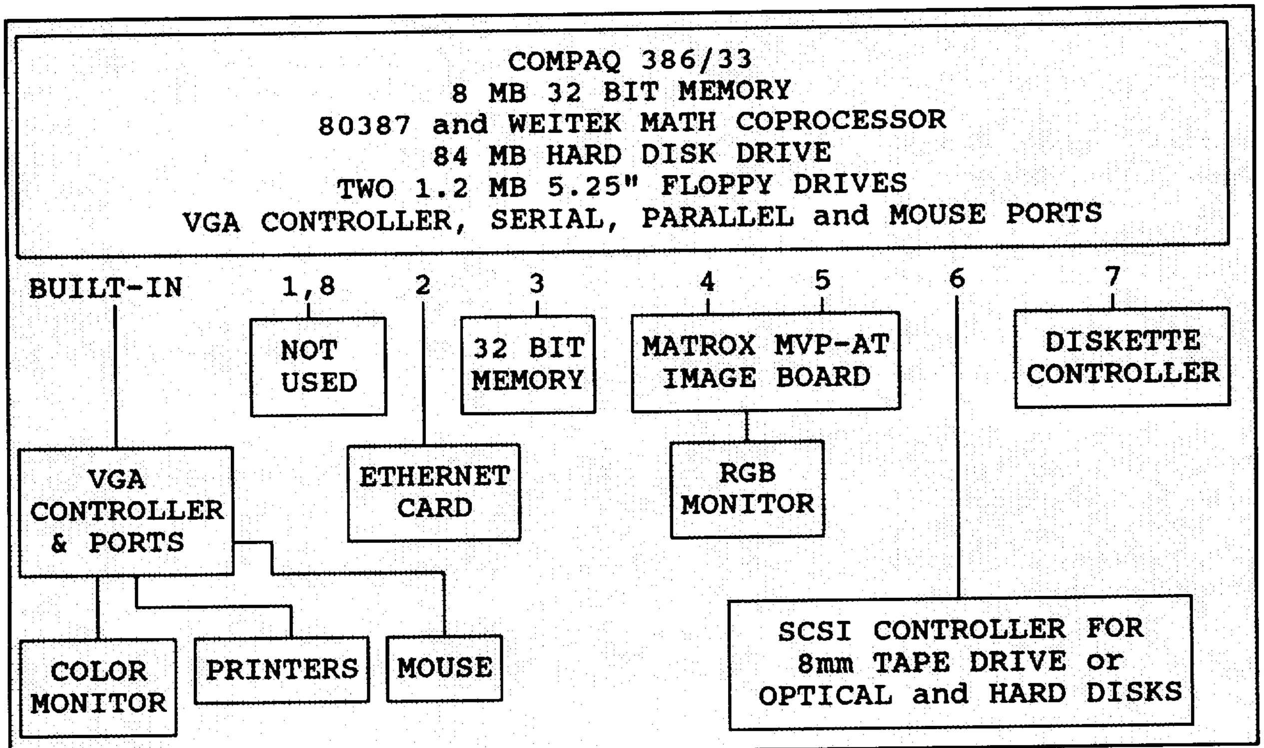

Figure 2. System configuration 2.

2. EXTERNAL PERIPHERALS

NEC Multisync II or 3D Color Monitor - This is a monitor which has a 14-inch diagonal display and a 13-inch viewing area. It is connected to and controlled by the AST-3G EGA control card or VGA controller and is used to display text and graphics and accept user input for all the utilities and applications.

Mitsubishi HA-3905L9 ADK Monitor - This is a 20-inch diagonal monitor which was connected to the MVP-AT image board. It is used for image and graphics display under PC-SEAPAK.

Imprimis WREN V Disk Drive - This is a high capacity, full-height, 5.25" Winchester disk drive. Its formatted capacity is 600 MB with an average seek time of 16 ms and a data transfer rate of 9-15 Mbits/sec on the SCSI interface. With the high capacity and fast access time of this disk, all the software utilities can be installed and all the software development can be done on it.

Summus GigaTape - This is an 8mm tape drive which is capable of backing up 2.33 gigabytes of data on one 8mm data cartridge. The GigaSAFE from Summus is a menu-driven software utility for backups or data interchange between PC and other host systems. The utility also supports two file formatting options, ANSI standard or Summus proprietary.

Summus LightDisk - This is a 5.25" erasable optical disk with an unformatted capacity of 650 MB (594 MB formatted) on one double-sided optical disk. The average access time is 90 ms and the data transfer rate is 620 KB/sec. This device can supply unlimited storage and is a good supplement to the storage on the hard disk and for backups.

Cipher M990 Tape Drive - This is a 9-track reel-to-reel tape drive. It is connected to and controlled by the Flagstaff tape controller card. Both 1600 and 6250 bpi tapes are supported by this system. For PC-SEAPAK, it is used to ingest the data from the CZCS level-1 and AVHRR level-1b tapes.

Hewlett Packard PaintJet Printer - This is a color graphics printer. It is connected to the parallel port on the multipurpose controller card. It can be used as the regular output device for text and graphics and is supported by the leading graphics and word processing software packages. The dithering methodology with the four ink cartridges (black, cyan, magenta, yellow) allows up to 330 colors. In PC-SEAPAK, there is a driver program which can print an image using 16 colors for the displayed image and 7 colors for the graphics overlay.

Microsoft Mouse - This is a two button mouse device. In the first (20 MHz CPU) configuration, it is connected to the serial port on the multipurpose controller card through a switch box; the second configuration has a built-in port for the mouse. The mouse is primarily used for controlling the cursor in PC-SEAPAK.

3. INSTALLATION CONFLICTS

Most of the controller cards and peripherals can be installed and connected easily with default setups by following the instructions in their manuals. The only problems with the Figure 1 and 2 configurations are conflicts by the default setups among the Ethernet card, the MVP-AT image board, and the SCSI host adapter. The first conflict is between the Ethernet card and the MVP-AT board where they both use the I/O address 300H. The second conflict between the MVP-AT board and the SCSI host adapter results from the fact that they both access the memory addresses D000 to DFFF.

To overcome these problems, the I/O address on the Ethernet card was changed to 200 and the memory address on the SCSI host adapter was changed to CC00. Since the MVP-AT image board is a two-board set with the dip switches for the I/O and memory addresses located inside, one needs to unscrew the board connections and divide two multiple pin connectors between the two boards in order to change its setups. This is not an easy task and should be done only if necessary.

There are always potential conflicts between controller cards. Normally, they occur on the I/O address or memory address setup,

but they can also occur with the DMA or interrupt channel selections. Usually, when there is a conflict, it will hang up the machine or fail to run some applications without any warning or error messages. To avoid potential conflicts between boards, especially when installing a new board or changing the setups of an existing board, always check the setups of the I/O address, the memory address, the DMA, and the interrupt channels. It is useful to keep a record of this information for each board.

4. MONITOR FOR MVP-AT

The MVP-AT image board can be configured through software to send different signals to the display monitor. Basically, the MVP-AT can send out two types of signals, interlaced and non-interlaced. For the interlaced signal, the frequency of the horizonal synchronization from the MVP-AT is 15.75 KHz and for the non-interlaced signal it is 31 KHz. The horizonal synchronization frequency determines the pixel resolution. For the MVP-AT, both the interlaced and non-interlaced options have a 512 pixel resolution. As for the frequency of the vertical synchronization from the MVP-AT, there are the 60-Hz American standard and the 50-Hz European standard options. For the American standard, only 480 lines can be displayed on the monitor, but for the European standard, 512 lines can be displayed.

To determine what kind of monitor and display format can be used with the MVT-AT, the frequency ranges for both horizontal and vertical syncs should be obtained from the monitor's manual. For example, the C-3920 from Mitsubishi Electric Corporation has a horizontal frequency range of 15.5 KHz to 23.5 KHz and a vertical frequency range of 40 Hz to 70 Hz. In this case, only the interlaced mode from MVP-AT can be selected since the non-interlaced mode needs a horizontal frequency of 31 KHz, which is out of the monitor's range. The vertical frequency in this monitor allows the MVP-AT to display 480 lines or 512 lines since its 40 Hz to 70 Hz range covers both 60 Hz and 50 Hz.

The default setup for the C-3920 is 60 Hz and 480 lines. To change it to 50 Hz and 512 lines, the standard line number for all PC-SEAPAK images, an adjustment must be made. The program SPKSETUP under the PC-SEAPAK directory is used to change the configuration file for MVP-AT setups. If the vertical frequency from MVP-AT does not match the vertical frequency on the monitor, the image will jiggle and will be unclear. The vertical synchronization can be adjusted by turning the V-HOLD knob until a steady display is obtained. Depending on the monitor, the V-HOLD knob may be located inside or outside of the monitor box. Unfortunately, the V-HOLD knob of the C-3920 is located inside the monitor box and the cover must be removed for the adjustment.

Another example is the Multisync II monitor from NEC. This monitor automatically adjusts the horizontal frequency from 15.5 KHz to 35 KHz and the vertical frequency from 50 Hz to 80 Hz. Tests on the interlaced and non-interlaced as well as with 480 and

512 line displays were successful. The only adjustment that may be required is with the vertical screen size so that the full image can be displayed within the screen range. This is easy to do for this monitor, since the V-SIZE knob is inside the front control box located just under the screen.

The new model HA3905L9 from Mitsubishi Electric corporation has a horizontal frequency range from 15.7 KHz to 35.5 KHz and a vertical frequency range from 45 Hz to 80 Hz as well as the auto-tracking capability. This monitor can be used on both interlaced and non-interlaced modes to display 480 or 512 lines on the screen without any adjustment.

1. PC-SEAPAK INSTALLATION

PC-SEAPAK should be installed on the hard disk with enough available storage space. The current version of PC-SEAPAK requires about 20 Mbytes of storage of which 94% are executable files for individual application programs and the rest are parameter files, configuration files and help files in text or binary format. In addition, 4.1 Mbytes of the CIA World Data Base, 3.7 Mbytes of the NASA Total Ozone Mapping Spectrometer (TOMS) data files and 4 Mbytes of the Navy bathymetry data base also need to be installed on the hard disk. In addition, HALO88 fonts are supplied on a separate disk through a licensing agreement with Media Cybernetics. These fonts are described in the Appendix. Also see Appendix for additional information on software and vendors.

PC-SEAPAK may be installed onto a hard disk from the distribution diskettes by following a simple series of steps. The installation procedure is as follows:

1. Create a SEAPAK directory (e.g., D:\SEAPAK) on the hard disk.
2. Copy all files except the HALO88 font and driver files, the CIA World Data base files, the Navy bathymetry data base file and the NASA TOMS data files from the PC-SEAPAK distribution diskettes to the SEAPAK directory on the hard disk.
3. Select or create a directory (D:\HALO88) and copy all HALO88's font and driver files from the distribution diskettes to that directory.
4. Select or create a directory (D:\CIADB) and copy all eight CIA World Data Base files from the distribution diskettes to that directory.
5. Select or create a directory (D:\PCTOMS) and copy all nine NASA TOMS data files from the distribution diskettes to that directory.
6. Select or create a directory (D:\BATHY) to contain the NAVY bathymetry file. Because of its size, the file was created with the BACKUP command and the RESTORE command must be used to restore it from the distribution diskettes to the root directory of device D. This requires that the file then be copied to the subdirectory and deleted from the root directory. For example,
   ```
   RESTORE A: D:\
   COPY D:\BATHY.DAT D:\BATHY\BATHY.DAT
   DEL D:\BATHY.DAT
   ```
7. Set up the environmental variable SEAPAK to point to the SEAPAK directory by typing "SET SEAPAK=D:\SEAPAK\" (the last character "\" is necessary) under the DOS prompt. This statement should be appended to the AUTOEXEC.BAT file in the boot directory so that the environmental variable SEAPAK will be assigned whenever you boot the system.

8. Check and modify, if necessary, the file SEAPAK.FIG under the SEAPAK directory. The description of the SEAPAK.FIG file is given later. In this step, you must make sure that the "LIBRARY" in the fourth line of the file is set to the SEAPAK directory created in step 1.
9. Run SPKSETUP by typing "SPKSETUP" in the SEAPAK directory to set up the memory and I/O addresses of the MVP-AT image board and the image display format. The program SPKSETUP is also used to identify the paths for the CIA World Data Base files, the NASA TOMS data files, the Navy bathymetry data file, and HALO88's font and device driver files. A detailed description of the program SPKSETUP is given in the Program Descriptions section of this guide.
10. Run INIT from the SEAPAK directory. This program must be run whenever the system is rebooted (hard or cold) so as to maintain the correct setups for the MVP-AT image board and its display format.

PC-SEAPAK may be started by changing to the SEAPAK directory and typing "SEAPAK". The PC-SEAPAK main menu should be displayed on the screen.

The SEAPAK.FIG file mentioned in step 8 is listed in Table 1 and contains the information needed for setting up the foreground and background colors for the screen, text, and fields, and defines the function keys, for all menu and input screens. In Table 1, the first line is the title information for PC-SEAPAK which will be displayed on the top line of all these screens. The second line specifies whether the display for these screens is in color or monochrome. The default "COLOR" in this line displays all the menu

Table 1. Contents of the file SEAPAK.FIG.

```
PC-SEAPAK                    VERSION 4.0 (OCTOBER 91)
COLOR
010 000 014 003 015 004 015 001 012 000 015 002 015 000 015 000 015 004
LIBRARY = D:\SEAPAK
HELP     0   59  F1          Displays help information for highlighted item
KEYS     0   60  F2          Displays a description of the command keys
CMD      0   61  F3          Invokes the command processor
SAVE     0   65  F7          Saves the parameter values in a save set
REST     0   66  F8          Restores the parameter values from a save set
BACK     0   67  F9          Moves back to the previous menu screen
NEXT     0   68  F10         Proceeds to the next menu screen or program
PGUP     8   73  PgUp        Moves to the previous page of multi-page menu
PGDN     8   81  PgDn        Moves to the next page of a multi-page menu
RIGHT    8    9  Tab         Moves to the array item on the right
LEFT     8   15  Shift-Tab   Moves to the array item on the left
END      0  113  Alt-F10     Ends current program and returns to top menu
EXIT    16   27  Esc         Exits from PC-SEAPAK
```

and input screens with colors defined by the color codes in the third line. If "UNKNOWN" is in the second line, the display for the screens will be in monochrome. The third line contains the color codes for the image and overlays of all items used for these screens. The fourth line specifies the location of the PC-SEAPAK directory and it should be changed if the PC-SEAPAK application programs are moved to another directory. The rest of Table 1 specifies the function keys used for all menu and input screens.

Table 2 lists the colors and color codes defined for the foreground and background of all items used in the menu and input screens.

In step 9, the program SPKSETUP will create two files under the SEAPAK directory specified in the input parameter SPKPATH, so the parameter SPKPATH must have the same directory name as created in step 1. The two files created by SPKSETUP will be named MVPAT.FIG and SPKPATH.PAR. The MVPAT.FIG file has only one line of information which contains two hexadecimal values for the memory and I/O addresses of the MVP-AT image board as well as three integer values--the pixel size, scan method and display mode--to set up the display format for the monitor. The default memory and I/O addresses are D000 and 300, the same as the default setups on the image board. If the memory or the I/O address have been changed on the board, the two values in MVPAT.FIG should also be changed by rerunning SPKSETUP or by editing the file. Depending on the display monitor, the MVP-AT image board can be set up with different display formats that depend on the selection of the pixel size (1, square pixels; 0, pixels with a 4:3 aspect ratio that is only valid for the interlaced scanning method), the scanning method (0, non-interlaced; 1, interlaced), and the displaying mode (0, 512 lines; 1, 480 lines).

Table 2. Color codes for menu and input screens.

Color	Code	Description
Light green	010	System title foreground
Black	000	System title background
Yellow	014	Main window foreground
Cyan	003	Main window background
White	015	Instruction window foreground
Blue	001	Instruction window background
Light red	012	Function key foreground
Black	000	Function key background
White	015	Highlight foreground
Black	000	Highlight background
White	015	Edited item attribute (edit menus)
Black	000	Default values attribute (edit menus)
White	015	Help window foreground
Red	004	Help window background

The SPKPATH.PAR file contains four lines of text that specify the directories of the CIA World Data Base, the Navy bathymetry data file, the NASA TOMS data files, and the HALO88 font files and device drivers used by PC-SEAPAK. In each of the text lines, the first 10 characters are the index name to be used by the PC-SEAPAK application programs to read the directory from the text specified after the tenth character. Currently, the index name "CIAWDB" is used for the CIA World Data Base files, "BATHY" is used for the Navy bathymetry data file, "PCTOMS" is used for the NASA TOMS data files, and "HALO88" is used for the HALO88's font files and device drivers.

2. HOW TO RUN PC-SEAPAK

PC-SEAPAK is a collection of independent programs. A PC-SEAPAK application program may be invoked through the PC-SEAPAK menu mode, the PC-SEAPAK command mode, or the DOS command mode.

2.1 PC-SEAPAK Menu Mode

To invoke the PC-SEAPAK menu mode, enter "SEAPAK" in the SEAPAK directory at the DOS prompt to get the main PC-SEAPAK menu, or enter "MENU" or "M" under the PC-SEAPAK command mode to restore the menu mode. On each menu screen, there are a number of items that may be selected. Each item, displayed in one row, represents a submenu or a program, the title information being on the left side and the submenu or program name on the right side. If the name is enclosed in brackets, it represents a submenu; if it is enclosed in parentheses, the item is a program. To make a selection, type a number or move the cursor with the <UP> or <DOWN> arrow key to the item you want to select and press the <ENTER> key. There are seven function keys defined for each menu screen:

<F1> - displays the on-line help information for the highlighted item (if it is a program not a submenu);
<F2> - displays a description of the command keys;
<F3> - invokes the PC-SEAPAK command mode;
<F9> or ALT<F10> - displays the previous menu;
<F10> - selects the highlighted item;
<ESC> - quits the PC-SEAPAK menu mode.

2.2 PC-SEAPAK Command Mode

To invoke the PC-SEAPAK command, press <F3> while in the PC-SEAPAK menu mode or type "COMMAND" when you are in the SEAPAK directory and at the DOS prompt. The PC-SEAPAK command mode prompt "SEAPAK>" will be issued. This mode allows the user to invoke any program directly by typing the program name followed optionally by the input parameters. This mode provides the more experienced users with a much faster method of invoking PC-SEAPAK programs. There are five formats for invoking a program in this mode:

```
format 1:   PROGNAME PARM1=VALUE1  PARM2=VALUE2 . . .
format 2:   PROGNAME VALUE1 VALUE2 . . .
format 3:   PROGNAME %SAVESET
format 4:   PROGNAME
format 5:   @SCRIPTFILE
```

where PROGNAME is a PC-SEAPAK program name.

In format 1, PARMn (n=1,2,...) are the parameter names defined in the first input screen of the program PROGNAME. For a list of those parameter names, type "HELP PROGNAME" in the PC-SEAPAK command mode and all the parameter names as well as their descriptions and default values will be displayed. This information is defined in the file with the same program name but with extension .PDF. When the parameter name is specified (format 1), PARMn= VALUEn pairs may be listed in any order. All parameters, except those for which default values are assigned, must be explicitly assigned values to bypass the input screen. In format 2, the parameter names are not specified but only the input values are listed following the program name. To skip the input screen, values for all the parameters including those having default values must be specified and must be in the same order as the parameters are listed on the input screen.

In format 3, SAVESET is a saveset name, i.e. a parameter file, which is a collection of all parameter values in the input screen. This file can be created only by the SAVE function key F7 when the input screen is displayed. In this format, all the parameter values will be assigned from the SAVESET file and the input screen will be skipped.

Note that, currently, only the PC-SEAPAK programs that have a parameter definition file (PDF) defined under the SEAPAK directory are able to run using formats 1, 2 and 3. The PDF files are ASCII files and have extension ".PDF".

In format 4, there is no argument after PROGNAME and the input screen will be displayed for the user to input values before the program is executed. This is the same as invoking the program from the PC-SEAPAK menu mode or from the DOS command mode (discussed below).

In format 5, the SCRIPTFILE is an ASCII file that contains a listing of command lines, in any combination of formats 1, 2, 3, or 4, to be executed in sequence. The user may use this format to set up a procedure for a demonstration or an unattended batch run. The "@" symbol is required.

Other commands in addition to the program invocation commands may be used in the PC-SEAPAK command mode. The command "HELP PROGNAME," mentioned above, is used to display the information for input parameters of the program PROGNAME. The command "MENU" or "M" is used to switch to the menu mode. The command "EXIT" returns the user to the DOS command mode.

2.3 DOS Command Mode

The DOS command mode is the mode under DOS control which allows the user to use all the DOS utilities and to run all the

commercial application packages. The PC-SEAPAK programs can also be run under this mode since each program exists independently as a DOS executable file. To invoke a PC-SEAPAK program under this mode, change to the SEAPAK directory and then type the program name at the DOS prompt. Although the programs can be run from any directory if the SEAPAK directory path is specified by the environmental variable PATH, the color display for the input screen as well as all on-line help will not be available.

2.4 Parameter Input Screen

For most PC-SEAPAK programs, input parameters must be specified by the user before the programs will start to run. In some cases, the user will be prompted for additional information during the programs' execution. Under PC-SEAPAK, there is always a parameter input screen associated with each input parameter set. Each parameter input screen has the PC-SEAPAK title, the input screen title, the input parameters, each with a brief description and a field to accept the input, and a list of the available function keys.

During a parameter input screen session, the user can only use the <UP> and <DOWN> keys to move the cursor bar (highlight on parameter field) to any parameter field and use <LEFT>, <RIGHT> and <BACKSPACE> to edit the input in the field. Note that the <ENTER> key will terminate the input screen session, so all parameter inputs must be entered before <ENTER> is pressed. The following function keys are defined for each parameter input screen:

<F1>	- displays the on-line help information for the highlighted parameter field;
<F2>	- displays a description of the input screen's function keys;
<F7>	- saves the parameter values on current input screen into a save set;
<F8>	- retrieves and displays parameter values on current input screen from a save set;
<F9>	- goes back to the previous input screen if there is one; otherwise same as ALT<F10>;
<F10>	- goes to next input screen if there is one; otherwise same as using the ENTER key;
ALT<F10>	- quits the input screen session and the program and returns to the previous mode;
<ESC>	- quits the input screen session and the program and returns to DOS command mode.

2.5 Memory Size

Although DOS allows 640 Kbytes of RAM memory to be used, due to the overhead of DOS itself and the space required for the environment variables, about 600 Kbytes are available for all other applications (under DOS 5.0 without memory resident programs or drivers). This amount of memory is further reduced by the addition

of device drivers for such things as a virtual disk, a mouse, the tape drive and memory resident programs such as Sidekick and DECnet-DOS. Also, DESQview requires a fair amount of memory. Since most device drivers are installed in the CONFIG.SYS file, the memory these and other memory resident programs occupy cannot be regained. The only way to clear these is to remove them from the CONFIG.SYS file and execute a cold reboot (turn the machine off, then on).

Running programs without enough memory may cause unpredictable results or hang up the machine. The DOS command CHKDSK may be used to check the amount of currently available memory and the command EXEMOD may be used to check a program's minimum load memory sizes.

Most of the PC-SEAPAK programs have memory load sizes (not the executable file size) of 200 to 400 Kbytes and should not have any problems executing even when some device drivers and memory resident programs are loaded.

3. ADDING ITEMS INTO PC-SEAPAK MENUS

In PC-SEAPAK, each menu can contain up to nine selection items. Each item can be an entry of an application program or a submenu. Each menu has a text file in the SEAPAK directory with the extension ".MNU" which is used by the PC-SEAPAK menu system to display the selection items of the menu on the screen.

Table 3 is the listing of the MAIN menu file presented here as an example. "$START" in the first line and "$END" in the last line are required to bracket the file's information. "$TITLE" in the second line specifies the menu title. "$ID" specifies the entry

Table 3. Listing of the file MAIN.MNU.

```
$START
$TITLE=PC-SEAPAK MAIN MENU
$ID=INGEST.MNU
CZCS and AVHRR ingestion                    [INGEST]
$ID=CZCSL2.MNU
CZCS level-2 processing                     [CZCSL2]
$ID=IMAGING.MNU
MVP-AT frame buffer programs                [IMAGING]
$ID=IMGFILE.MNU
Image file information                      [IMGFILE]
$ID=GEOGRAPH.MNU
Geographic programs                         [GEOGRAPH]
$ID=MATH.MNU
Mathematical programs                       [MATH]
$ID=UTIL.MNU
Utility programs                            [UTIL]
$END
```

Table 4. PC-SEAPAK menu tree structure.

MAIN	- INGEST	- CZCSIN
		- AVHRRIN
	- CZCSL2	- ATMOS
		- L2PROD
	- IMAGING	- INITIAL
		- FRMBUF
		- OVERLAYS
		- LUTCOLOR
		- MOSAIC
	- IMGFILE	- DATA
		- HEADER
	- GEOGRAPH	- PROJECTN
	- MATH	- HARDFCT
		- SOFTFCT
		- STAT1
		- STAT2
	- UTIL	- HARDCOPY
		- MIAMI
		- VAXTOPC

file of an application program or a submenu. The entry file must be an executable file (".EXE" file) or a batch file (".BAT" file) or a menu file (".MNU") in the SEAPAK directory. Each "$ID" line is followed by a description line that will appear on the menu screen and will be numbered consecutively from 1. The entry file name (without the extension) is written a the end of each corresponding description line. By convention, this name is enclosed within square brackets if it represents a submenu or within parentheses if it represents a program.

Table 4 is the current PC-SEAPAK menu tree structure. It is easy to add any application programs or submenus into the PC-SEAPAK menu system. First, select a menu file into which you want the new entry (a program or a submenu) to be added. Second, use any editor to add the "$ID" and description lines into the appropriate location in that file. If the new item to be added is a submenu, a menu file with extension ".MNU", having the same format as that of MAIN.MNU (Table 3), should also be created for that submenu. All entry programs specified by the "$ID" statements must exist in the SEAPAK directory. If the executable file is not in that directory, the user can create a batch file in the SEAPAK directory and use it to start any program outside the SEAPAK directory.

4. PC-SEAPAK SPECIFIC TOPICS

Frame Buffer: Under PC-SEAPAK, the 1 Mbyte of video RAM on the MVP-AT board is configured as four 512x512x8 frame buffers numbered 0, 1, 2 and 3. Normally, frame buffers 1, 2 and 3 are

used for image display and frame buffer 0 is reserved as an overlay frame buffer (discussed later) to display the cursor, the menu of function keys and other overlay graphics.

Look-up Tables: In the MVP-AT, there are two look-up table sets, the input look-up table (ILUT) and the output look-up table (OLUT). Each of the sets contains 32 palettes numbered from 0 to 31 and each palette has its own red, green and blue look-up table of 256 locations. In this guide, all the palettes mentioned will refer to the output look-up table unless otherwise noted.

Palettes: As discussed above, there are 32 palettes numbered 0 to 31 available for the ILUT and OLUT. Each palette has its own red, green and blue look-up table, each of the look-up tables has 256 locations, and each location can have a value of 0 to 255. In most situations for PC-SEAPAK, only the palettes of OLUT are used for pseudocoloring an image and overlaying graphics on an image. Some of the palettes are already reserved for special purposes. Palettes 11 to 14 are used for look-up tables to display images without overlays on frame buffers 0 to 3. Palette 0 is used for the look-up table of the displayed image with an overlay display. Palettes 1 to 7 are used for overlay graphics. Palette 8 is used for the cursor. Palettes 17 and 18 are used for the display of function key menu.

Image Display without Overlay: All four frame buffers may be used to display an image if no overlay is needed. In this case, palettes 11 to 14 in the OLUT are used for the default look-up tables to display images on frame buffers 0 to 3.

Image Display with Overlay: When an image is being displayed, another frame buffer is required to display an overlay. In PC-SEAPAK, frame buffer 0 is reserved for the overlay frame buffer. The eight-bit depth of each pixel in the overlay frame buffer can be separated into two sets, the four least and the four most significant bits. The two four-bit selections are used to select overlay palettes 0 to 15 (the four least significant bits) or 16 to 31 (the four most significant bits). Only one set can be active at one time. In PC-SEAPAK, the least significant four bits (palettes 0 to 15) are used for overlay graphics and the most significant four bits (palettes 16 to 31) are used for the overlay display of the function key menu.

When overlays are used, every pixel in the displayed image frame buffer has its own palette for showing the color on the monitor. The palette for each individual pixel in that frame buffer is determined by the value of the least significant four bits (0 to 15) at the same pixel location as for the overlay frame buffer. Once the palette is determined, the background pixel value (0 to 255) will be used as the index for the intensities of the red, green and blue colors from the palette's corresponding look-up tables.

In PC-SEAPAK, only the seven palettes 1 to 7, of the 16 available, are used for the overlay graphics. The red, green and blue look-up tables for these palettes should always have constant values in each of their 256 locations so that the overlay graphics of a specified palette always have the same color on the displayed image regardless of that image's pixel value. If any of the 256 locations in the look-up tables are not constant, the overlay color on the displayed image may not be same. This is because the pixel value in the frame buffer of the displayed image is used as the index of the color intensities in the look-up tables. Palette 0 is used for the look-up table of the displayed image whenever the overlay is active. In PC-SEAPAK, if there is an overlay, the look-up tables of palette 0 have to be loaded from palettes 12, 13 or 14 which are reserved for image frame buffers 1, 2 and 3 for non-overlay displays.

Cursor: The MVP-AT does not support a hardware cursor device. The cursor used in PC-SEAPAK is designed and controlled entirely by the software and the mouse device. The default cursor is a "+" of size 15x15 pixels and is displayed through the overlay by using the OLUT palette 8 with red as the default color. The default cursor setup can be changed by editing the file SPKDEF.PAR in the SEAPAK directory.

Menu of Function Keys: For most of the PC-SEAPAK programs, function keys are defined for various program functions. For the user to review which function keys are defined and what their assignments are, the function key menu is used. By default, the menu is overlaid on the image frame buffer with default OLUT palette 17 for the foreground and OLUT palette 18 for the background. The default palettes setups may be changed by editing the file SPKDEF.PAR, but must be palettes from 17 to 31. Because the most significant four bits in the overlay frame buffer are used to display the function key menu, the overlay graphics which use the lower four bits will be inactivated. Thus, the menu should always be toggled off after reviewing its text. The mouse's right button and the ALT<F1> key may be used for toggling the display of the function key menu on and off.

File Name Structure under PC-SEAPAK: In the current PC-SEAPAK version, all input and output file names have the same structure as under DOS except that the maximum number of characters cannot exceed 40. The file names should be of the form DRIVE:\PATH\FILNAME.EXT, where DRIVE is the one character drive name, PATH is the directory path, FILNAME is a name of up to 8 characters, and EXT is the extension of the file name and is usually three characters long. The current drive and directory will be used if DRIVE and PATH are omitted. There are no default strings for EXT and FILNAME. Note that some programs (e.g., TP2IMG, TP2DSK, L2MULT) that create output files may use FILNAME as the root and append a one character index to it. In such cases FILNAME should not have more than seven characters.

5. RUNNING PC-SEAPAK UNDER DESQVIEW

Quarterdeck Office System's DESQview is a multitasking and window environment which allows programs to run simultaneously within different windows. PC-SEAPAK may be installed as an item under DESQview's open window menu and invoked by selecting that item. Alternatively, it may be invoked from DESQview's DOS window which must be installed separately on DESQview's open window menu. Also, any PC-SEAPAK program may be installed as an individual item under DESQview's open window menu.

The steps for installing an item on DESQview's open window menu can be found in DESQview's manual and will not be discussed here. There are only two things which need to be considered for the installation of PC-SEAPAK under DESQview: the memory size for running the application program in a window and the program name to start the application. The maximum available memory size under DESQview can be found by using the Memory Status utility supplied by DESQview. This maximum should be used for all the windows running PC-SEAPAK or individual PC-SEAPAK programs. "D:\SEAPAK\SEAPAK" should be used to start the SEAPAK window, and "D:\SEAPAK\PROGNAME.EXE" should be used to start any of the individual PC-SEAPAK programs in a window, where "D:\SEAPAK" is the SEAPAK directory and "PROGNAME" represents one of the specific program name. Also, the SEAPAK directory D:\SEAPAK should be put into the directory field when you install PC-SEAPAK or individual PC-SEAPAK programs into the open window menu. As for the DOS window, any user specified batch file or DOS command can be used to start it. Running programs in the DOS window requires that the DOS window has a sufficient memory allocation.

All PC-SEAPAK programs except the tape ingest programs and the programs that need more memory than DESQview's maximum available size may run under DESQview. The user may use the DOS utility EXEMOD to check the memory load size (not file size) for each program.

6. RUNNING PC-SEAPAK PROGRAMS UNDER DOS PROTECTED MODE

Due to the memory limitation (640 KB) under the DOS real mode, some of the PC-SEAPAK programs such as L2MULT, MAPIMG, BHL2MULT, BHMAPIMG and FILLA that were developed under Microsoft Fortran may have problems being loaded or executed when memory resident programs or device drivers take up excessive amounts of memory. To solve the problem, those programs were converted to run under 386's protected mode with MicroWay's NDP Fortran compiler and Phar Lap's Link and DOS-Extender. The NDP Fortran compiler is a 32-bit compiler that can generate code for the Intel 80387 and Weitek floating-point coprocessors. Phar Lap's Link is a linker that generates executable code to run under 386's protected mode and the DOS-Extender is a program required to load those programs and run under protected mode.

The programs running under 386's protected mode do not have the DOS 640 KB memory limitation. Their limit is the total memory size of the system. Since the DOS-Extender run-time version was purchased, it has been integrated into those programs so that users do not need to buy it separately. Currently, only the following programs are set up to run under protected mode: L2MULT, MAPIMG, BHL2MULT, and BHMAPIMG for the Intel 80387 coprocessor, WTKLM, WTKMP, BHWTKLM and BHWTKMP for the Weitek coprocessor, and FILLA and EOF for both the 80387 and Weitek coprocessors. The Weitek version of programs have the same functions as the Intel 80387 versions but execute about twice as fast. Of course, the system must have a Weitek coprocessor installed for these programs to be used. Most manufacturers provide a Weitek socket on the mother-board.

SUMMARY OF MAJOR PROCESSING STEPS

A typical series of major processing steps that a user may follow in order to process a CZCS or AVHRR scene are presented in Fig. 1. Many of the considerations involved for each of these steps are discussed in detail in following sections of this chapter. Detailed information about individual programs is given in the corresponding sections of PC-SEAPAK programs.

PC-SEAPAK programs are generally independent of each other and the order of their appearance on menus or in Fig. 1 does not imply a forced execution sequence. Although a user's analysis will normally follow a logical progression of steps such as those suggested by Fig. 1, the large number of programs available allow numerous options throughout the analysis approach.

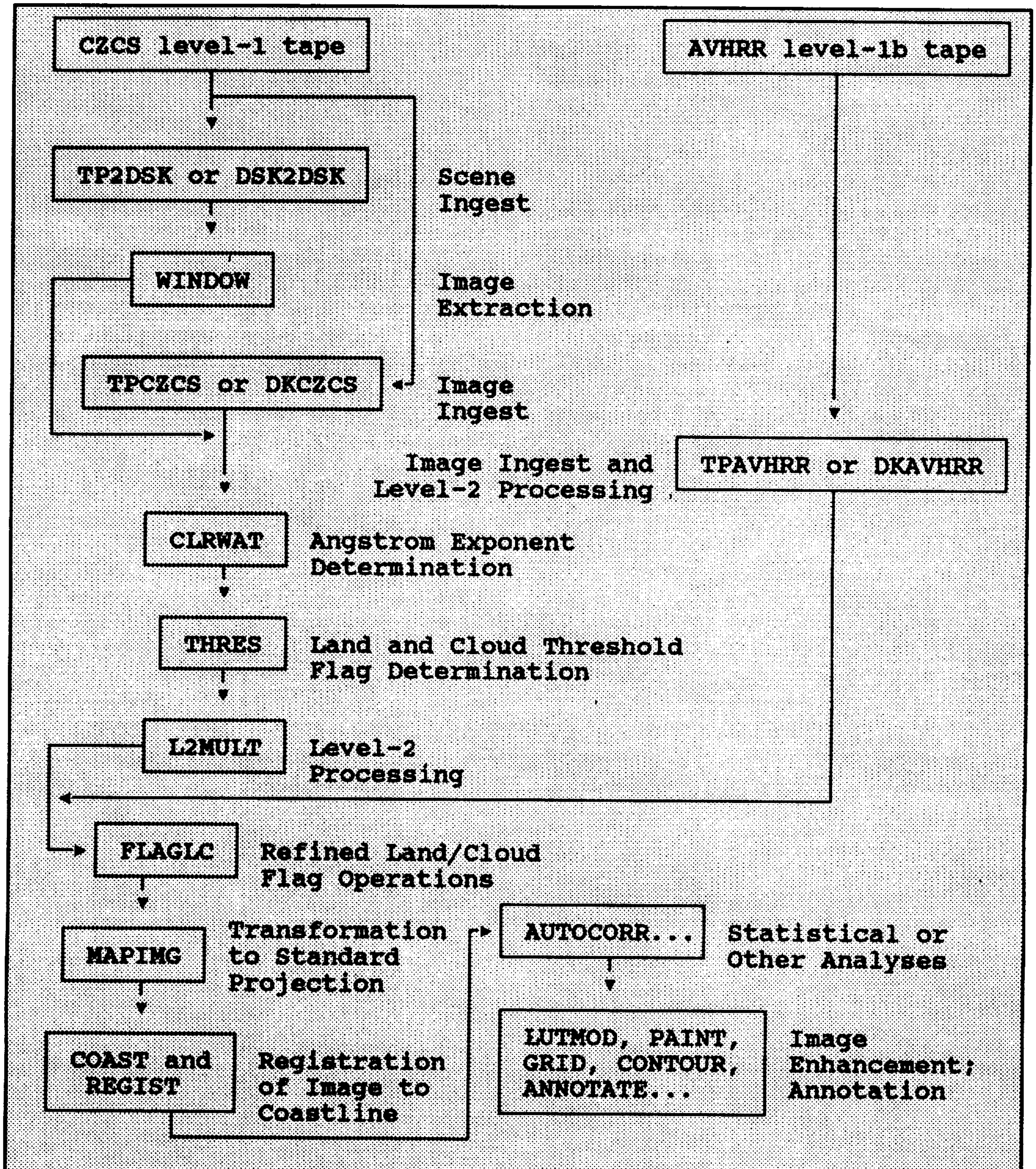

Figure 1. Major processing steps for typical analyses of CZCS and AVHRR images using PC-SEAPAK programs.

INGESTING LEVEL-1 DATA

In this section, the generation of CZCS level-1 images from tape and disk files will be discussed. The creation of level-1 image files from AVHRR tapes is similar to the procedure for CZCS data, but with some important differences. Unlike CZCS files, AVHRR tape files from NOAA can be any number of scan lines. The CZCS level-1 files normally have 970 scan lines per file, but may have fewer. Also, the AVHRR local area coverage (LAC) data has 2048 and global area coverage (GAC) data has 409 samples per scan line whereas a CZCS scan line has 1968 samples. Sea-surface temperature (SST) level-2 products are created when AVHRR data is ingested, while level-2 products from the CZCS must be generated separately (see the section on generating CZCS level-2 products).

The programs TPCZCS, TPAVHRR, DKCZCS and DKAVHRR are used to create 512x512 image files from tape and disk scene files of CZCS and AVHRR LAC and GAC data. However, TP2DSK and DSK2DSK may also be used for CZCS data to ingest up to three complete 970x1968 scenes from a tape (TP2DSK) or from disk (DSK2DSK) as one set of files from which 512x512 image files may be extracted using the program WINDOW. This approach is often more convenient than using TPCZCS or DKCZCS since it permits the user to visually determine the location of the extracted image in the overall scene.

Note that all tape ingest programs in the current version of PC-SEAPAK can work only with the M990 GCR tape drive from Cipher Data Products, Inc., and Flagstaff Engineering's tape utilities. The programs DKCZCS, DKAVHRR and DSK2DSK are disk versions of TPCZCS, TPAVHRR and TP2DSK that ingest data from a disk instead of from tape if the tape data is first copied onto the disk.

<u>Using TPCZCS</u>: The program TPCZCS is used to ingest level-1 CZCS scenes and create 512x512 pixel images. In addition to the creation of full-resolution images, positive reduction factors may be used to subsample the data, since the level-1 scene may have up to 970 scan lines of 1968 samples, and negative reduction factors may be used to create images magnified by pixel duplication. Using TPCZCS requires a little arithmetic unless one assumes the file is a full scene and an overview is desired (reduction factors 4 and 2 for samples and scans, respectively). Usually, a user will want to generate a set of overview images in order to see the full scene. From the overview images, the coordinates of subimages can be determined and they are usually used in the Angstrom exponent determinations required by the level-2 programs discussed in the section on generating CZCS level-2 products.

In PC-SEAPAK, image files of each CZCS band are created independently of the others. Systems like the University of Miami's DSP interleave the data from each channel resulting in one file per scene(s). SEAPAK creates six image files and one control point file which contains the navigation data. In TPCZCS, the user provides a root name and extension for the image files, and the

band numbers are appended to the root name automatically. The control point file is simply the root name with a ".CTL" extension (rootname.ctl). Each convention has its advantages and disadvantages. Separation of the image files allows users to easily delete unnecessary files such as IR-band images thus saving disk space, but requires more file management.

In TPCZCS, the WINDOW parameter values determine the start and end pixel and line coordinates. These must match the reduction factors (REDFAC values) used. See the TPCZCS program section for more information.

Using TPAVHRR: This program is used to ingest an AVHRR scene of HRPT, LAC or GAC data from a tape in the format of those generated by NOAA/NESDIS/NCDC/SDSD (Kidwell, 1988) as well as to generate the SST image. The data must be in packed format, with time incrementing, and be a full set copy (as opposed to selective extract subsets where certain channels are selected). Unlike CZCS scenes, the AVHRR scene in this format may contain variable scan line numbers. The program allows the user to scan the tape scene first, if the user has no information of the input scene, in order to get the starting and ending scan line numbers for the windowing (WINDOW) and reduction factor (REDFAC) selections. Also, since the AVHRR scenes may be obtained while the satellite is ascending (flying south to north) or descending (north to south), the enumeration of the samples and scan lines may be reversed. See the TPAVHRR program section for more detailed information.

Using TP2DSK and WINDOW: Rather than ingesting reduced-resolution CZCS scenes one at a time to obtain an overview and then reading in subscenes for merging as described above, the combination of the programs TP2DSK and WINDOW may be used. For TP2DSK, all one needs to specify are the sequence number of the first file and the number of files to be read from the tape. The files must be consecutive satellite scenes as they normally are on tapes obtained from NASA or NOAA. The output is one large file containing the scenes' data at full-resolution. If significantly large gaps exist between the scenes, execution will not be completed. Currently, up to three files can be ingested in this way.

The program WINDOW is very easy to use. The overview of the entire data file will be displayed at reduced resolution in one of the MVP-AT frame buffer along with a default window box. Then, using the function keys, the user can change the box size which will have dimensions that are multiples of 512 in each direction. Single images from one CZCS band or all the bands can be generated that correspond to the current box position. When the appropriate function key is depressed, the user will be prompted for the filenames required. The program automatically uses the boxed area to create a 512x512 sized image.

Using DKCZCS and DKAVHRR: The programs DKCZCS and DKAVHRR are the disk versions of TPCZCS and TPAVHRR and ingest CZCS level-1 data and AVHRR LAC and GAC level-1 data files from disk instead of from

tape. TPCZCS and TPAVHRR only supports the Cipher's M990 tape drive with Flagstaff Engineering's tape drive software. Users with different tape drives for their PCs or on different systems must copy these files from tape to disk using the tape utilities supported by the tape drive, or download the files to their PC's from another system, before using the disk version programs.

These programs support two types of disk formats, variable-record length and fixed-record length, for CZCS and AVHRR data. For CZCS level-1 tapes, each scene contains a header file and a data file. Since only the data file is needed, the user must skip the header file when copying the data from tape to disk. In the data file, the first and last records are documentation records and the records in between contain up to two minutes (970 scan lines) of CZCS radiance data. For the variable-record length format, documentation records are 5,328 bytes and the scan-line records are 12,780 bytes, whereas for the fixed-record length format, all records are 12,780 bytes.

For an AVHRR LAC tape, each file has a TBM (Terabit Memory) header as the first record, a data set header as the second record, a dummy third record, and a variable number of data records (fourth to end). Each scan line is contained in two data records. The variable-record length format has lengths of 122 bytes for the TBM header record and 7,400 bytes for the data set header, dummy, and data records; the fixed-record length format uses 7,400 bytes for all records.

For an AVHRR GAC tape, each file has a TBM (Terabit Memory) header as the first record, a data set header as the second record, and a variable number of data records (third to end). Each data record contains two scan lines. The variable-record length format has lengths of 122 bytes for the TBM header record and 6,440 bytes for the data set header and data records; the fixed-record length format uses 6,440 bytes for all records.

Depending on the tape utilities and the system, the user may select the variable-length or the fixed-length format for copying the tape data to disk. At the NASA/GSFC Laboratory for Hydrospheric Processes, we have used CZCS level-1 and AVHRR LAC and GAC files with variable-record lengths created with Cipher's M990 tape drive and Flagstaff Engineering's tape utilities on our PC and fixed-record length files created on a MicroVAX II and downloaded to our PC.

Using DSK2DSK: DSK2DSK is the disk version of TP2DSK which can generate full-resolution CZCS level-1 file from multiple disk scene files. The two disk file formats described in the previous discussion of DKCZCS are also supported by DSK2DSK.

GENERATING CZCS LEVEL-2 PRODUCTS

Presently, PC-SEAPAK has several programs in its CZCS level-2 menu, CZCSL2, and submenus that include a number of tools to assist the user in determining the input parameters required to generate level-2 products. These programs include L2MULT, CLRWAT, BXCLRWAT, L2BOX and THRES. L2MULT creates images of subsurface water radiance (or, optionally, normalized water-leaving radiance) at 443nm, 520nm and 550nm, aerosol radiance at 670nm, pigment concentration, Rayleigh radiance at 443nm (a useful diagnostic quantity when other standard products seem unreasonable) and diffuse attenuation at 490nm. In this discussion, the term "water radiance" will apply to subsurface water radiance, while water-leaving radiance and normalized water-leaving radiance refer to radiances just above the air-water interface. These will be discussed in more detail later. The multiple scattering model of Gordon et al. (1988) is used in all the programs since it is the most sophisticated model available for CZCS analysis and is the model used in the CZCS global processing project (Esaias et al., 1986; Feldman et al., 1989).

L2BOX allows the user to roam a level-1 scene using the cursor and compute the values of the level-2 products and additional quantities within a box. It provides the flexibility of changing input parameters such as the calibration correction factors, the aerosol correction parameters and the ozone optical thicknesses.

The primary parameters required for generating level-2 products are the aerosol correction parameters (the Angstrom exponents or epsilons) and the land/cloud flag threshold. CLRWAT, BXCLRWAT, SCREEN, and ANGST are designed to help determine the aerosol correction parameters while THRES may be used to fine tune the land/cloud flag.

Other parameters, such as the calibration correction factors and the ozone optical thicknesses, can also be varied. However, the use of calibration correction factors other than the default values is not recommended since such factors are not easily determined unless additional field observations of upwelling water radiance are available. Likewise, alternate ozone thicknesses should be used with caution since the default thicknesses are derived from the Total Ozone Mapping Spectrometer (TOMS) data for the time and location of the scene in question.

Early in the CZCS mission, the Nimbus Experiment Team and others realized that the sensor's sensitivity was degrading and that the rate of degradation rate was different for each of the 443nm, 520nm and 550nm bands. However, quantification of the degradation was difficult (Viollier, 1982; Gordon et al., 1983a; Hovis et al., 1985; Mueller, 1985; Gordon, 1987). Because of the relatively large atmospheric contribution to the total observed radiances (Gordon, 1981) and the great sensitivity of the bio-optical algorithms to the estimated water-leaving radiances (Clark, 1981), small errors in the calibration can induce sizable errors in

the derived geophysical products rendering them useless for many applications. Thus, a comprehensive investigation of the calibration over the entire period of sensor operation was undertaken during the global reprocessing of the CZCS dataset (Feldman et al., 1989).

By processing large quantities of clear-water imagery, R. H. Evans (unpublished) was able to develop a 'vicarious' calibration that was used in the global processing of the entire CZCS data set. Specifically, the normalized "clear-water" radiances (Gordon and Clark, 1981), $[L_W]_N^{443}$, $[L_W]_N^{520}$ and $[L_W]_N^{550}$, were assumed to be 1.40, 0.48 and 0.30 $mW/cm^2 \cdot \mu m \cdot sr$, respectively, the Ångström exponents were assumed to be 0 and certain geographical regions such as the Sargasso Sea were assumed be "clear-water" sites (pigment concentrations < 0.25 mg/m^3) at all times. Under these assumptions, analyses of the derived $[L_W]_N$'s indicated what calibration adjustments were required to produce the nominal "clear-water" normalized radiance values. The vicarious calibration of the 443nm band is somewhat tenuous because of the great variability in $[L_W]_N^{443}$ even in "clear-water". This calibration is called the "Evans" calibration in the level-2 processing programs.

<u>Step 1 - Determining the Angstrom exponents and the land/cloud flags</u>: In the global CZCS processing mentioned above, Angstrom exponents equal to 0 are used on all scenes and no attempt is made to compute Angstrom exponents for each scene. The terms Angstrom exponent and epsilon are both used interchangeably in discussing the aerosol correction. The reader is referred to Gordon et al. (1983) for a discussion of the terms which are related by the following equation:

$$e(\lambda) = (\lambda/670)^{n(\lambda)}$$

where λ is the wavelength (443, 520 or 550), and $e(\lambda)$ and $n(\lambda)$ are the epsilon and Angstrom exponent, respectively. So, for an epsilon equal to 1, the Angstrom exponent is 0. As epsilon increases, the Angstrom exponent becomes more negative. One of the assumptions in the atmospheric correction algorithm is that the aerosol radiance at 670nm is related to the aerosol radiances at 520nm and 550nm through this equation. Another assumption is that these do not change within a scene. The Angstrom exponent at 443nm is taken to be the average of the values at 520nm and 550nm because the water radiances at 443nm are too variable even in clear water for stable estimates to be derived.

In areas dominated by marine haze such as in the central gyres and along the western continental margins, Angstrom exponents of 0 (or epsilons of 1) are usually adequate. However, in regions influenced by continental haze such as the eastern U.S. coast and the Mediterranean Sea, these values often fail to remove the haze resulting in contamination of the level-2 products by underestimating the aerosol radiance. This produces an overestimation of the water radiance and an underestimation of pigment concentration. On the other hand, high concentrations of dust are often encountered

in the eastern tropical Atlantic Ocean, the western Pacific Ocean, and the Arabian Sea. These conditions can cause the 670nm band to saturate making an atmospheric correction impossible. The HAZE parameter (of program L2MULT, for example) has a default value of 254 which flags all saturated pixels, although some conditions may require the flag to be set lower. Dust contaminated data usually require Angstrom exponents greater than zero. One should always compare the water radiance and pigment images with the aerosol radiance or level-1 670nm radiance image to determine if features are correlated. If the haze is correctly removed, there should be no correlation.

When dense or continental haze is present, the user may try CLRWAT or BXCLRWAT in an attempt to find a better set of Angstrom exponents. CLRWAT and BXCLRWAT are described in the PC-SEAPAK programs section. Both CLRWAT (automated mode) and BXCLRWAT (interactive mode) use a set of criteria to eliminate pixels from consideration. Some of these criteria may be adjusted by the user. Solar zenith and spacecraft zenith angles are examples. If the sun is too low in the sky, the radiative transfer models may not work well enough for the estimation of the Angstrom exponents. Aerosol radiance is another example. If the haze is too dense or too small, the estimation of the Angstrom exponents will not be valid for the rest of the scene. Also, pixels which fail the land/cloud flag are eliminated. Defaults are provided for all these. In BXCLRWAT, the user roams the scene with a box cursor looking for the set of Angstrom exponents associated with the lowest value of a quantity called CLOW. CLOW is the ratio e(443)/La(670) where La(670) is the aerosol radiance.

This procedure was developed by the Nimbus Experiment Team and is presented in Williams et al. (1985a and 1985b), but the rationale behind it was never discussed. It has been used by SEAPAK users for several years and has been found to yield consistent and quite acceptable results (see Barale et al., 1986, and McClain et al., 1988). The best locations to search are those with very low pigments because the 670nm radiances will not be affected by the ocean's reflectance. Care must be taken to avoid pixels affected by sensor ringing on the down-scan side of bright areas such as clouds (Mueller, 1988; see the help text for the program RING), an effect that is most noticeable in the 670nm image. In addition, fringe areas around clouds that are not flagged by the land/cloud threshold can cause erroneous estimates of CLOW.

In CLRWAT, the user sets the maximum pigment threshold in order to ensure that clear-water pixels are used. The program then computes the epsilon frequency distributions at 443nm, 520nm and 550nm from all valid pixels. Certain statistics are derived from the frequency distributions and may be output to a text file. These include the minimum, the maximum, the mean, the median, the mean of the lowest 10%, the standard deviation and the interquartile range. From these, the user can select a set of epsilon values to use. CLRWAT also creates a special image whose pixel values indicate the rejection criterion for invalid pixels as well as the clear-water

pixels that passed all criteria. This special image may be displayed using the program SCREEN to color the various pixels according to each pixel's category.

When using CLRWAT or BXCLRWAT, care must be taken to set the cloud threshold properly. The default works in most situations, but care must be taken to avoid thin clouds and areas where the 670nm radiances are saturated. Band 5 (750nm) was designed for land/cloud identification. Clouds at low solar elevations tend to be less bright, so if the solar zenith angle is high (the program DMPHDR can be used to find out) or if there are a lot of thin clouds, the default value of 21 counts (gray levels) in the 750nm image may need to be reduced. THRES or READ can be used to determine the "best" threshold. In general, it is advisable to use the level-1 670nm image for BXCLRWAT because it is sensitive to haze, clouds and ringing.

If CLRWAT or BXCLRWAT do not yield useful results, ANGST is often helpful. ANGST is based on a technique developed by Arnone and LaViolette (1984) and is designed to allow the user to interactively remove haze from the level-1 443nm, 520nm and 550nm bands using the 670nm band as the reference aerosol band. In ANGST, one operates on each band separately using the cursor to fine tune the haze removal. This is particularly useful when there is a specific haze feature to be removed. In this way, incremental increases in the Angstrom exponents can be made until there is no evidence of the feature left in the water radiance images. The program allows the user to check the water radiance values in the scene using the cursor and to stretch the image contrast in order to see more clearly the features. In using the level-2 programs which require Angstrom exponents, the same Rayleigh scattering model must be used as was used in determining the Angstrom exponents.

Step 2 - Using L2MULT: (See the detailed descriptions of L2MULT, and other programs of the L2PROD menu, in the program sections for more information on options and parameters.) Once the Angstrom exponents and the land/cloud flag are determined, L2MULT is used to generate the level-2 products. In L2MULT, the multiple scattering model of Gordon et al. (1988) is used. The program is designed to provide as much flexibility as possible in the selection of algorithms and input parameters. An inexperienced person should stick with the defaults provided for parameters such as the ozone optical thicknesses, the calibration algorithm, the pigment algorithm, water radiance range and the method used for the aerosol correction (ITERATE is the selection parameter).

The user is given the choice of generating subsurface upwelling water radiance or normalized water-leaving radiance (Gordon and Clark, 1981) images. The parameter is NORMWAT. The transformation from subsurface water radiance to water-leaving radiance is a function of the Fresnel reflectivity and the index of refraction. However, it is wavelength independent and cancels out when ratios are used in the pigment algorithm. Normalized water-leaving radiances have the solar zenith angle dependence removed and therefore have the advantage of being nearly constant at 520nm

and 550nm in clear-water regions. The normalized 443nm water-leaving radiances are variable because of the great sensitivity to pigment concentration at 443nm, even in clear water regions. At this time, algorithms for deriving other quantities from normalized radiances have not been developed. Finally, the user is given the option of applying the Smith and Wilson (1981) iteration method for computing the water and aerosol radiance fields. This option creates an eighth output field for water radiance at 670nm since it does not assume that water radiance at 670nm is zero as does the Gordon et al. (1983) algorithm. For each pixel, if the algorithm does not converge after 10 iterations, it is assumed to be an invalid pixel and a 0 gray level (black) is assigned to all output images at that pixel.

One input which may require some advance consideration is the cloud flag. In L2MULT, the 443nm level-1 radiances are used with the 750nm (band 5) radiances to discriminate land from clouds if the MASKLC parameter is set to "1." In this case, pixels which are brighter than the CLOUD threshold gray level value and which are also flagged by the LANCLD threshold will be set to a 255 gray level in the level-2 images. Pixels which fail the LANCLD threshold, but pass the CLOUD threshold are set to a 0 gray level value. If the CLOUD threshold used results in black areas over the ocean, these pixels may be changed to 255 using the program RESCALE. However, RESCALE operates on the entire image. It is assumed that a blotch of the land areas will be overlaid on the image, so it is not essential that all land be flagged black by the level-2 processing.

There are two pigment algorithms from which to select. The parameter is PIGMENT. One is the standard, two-channel "branching" algorithm of Gordon et al. (1983). This algorithm switches from an equation based on water radiances in the 443nm and 550nm bands to one based on the 520nm and 550nm bands once the concentration reaches 1.5. In this case, PC-SEAPAK also switches to the 520nm-550nm equation once the 443nm subsurface water radiance drops below 0.15. This is done because of imperfections in the switching logic which occasionally allow the 443nm-550nm combination to be used even though it yields values much greater than 1.5. The exact algorithm as implemented in PC-SEAPAK is

1. if Lw(550)=<0, then P=46.34456 (saturated); else,
2. if Lw(443)>0.15,
    then P = A2 * (Lw(443)/Lw(550))**B2, (A)
    where $\log_{10}$(A2)=0.053 and B2=-1.71;
    if P>=1.5 and Lw(520)>0
        then P = A4 * (Lw(520)/Lw(550))**B4, (B)
        where $\log_{10}$(A4)=0.522 and B4=-2.44;
        if P<1.5, then use (A) above;
3. if Lw(443)=<0.15 and Lw(520)>0, then use (B) above;
4. if Lw(443)=<0.15 and Lw(520)=<0,
    then P = 46.34456 (saturated);

where Lw represents the water-leaving radiance for the band of the specified wavelength (nm) and P is the pigment concentration in mg/m^3.

The other algorithm uses a three-channel equation provided by Dennis Clark (see Muller-Karger et al., 1990):

```
1. if (Lw(550)>0) and (Lw(443)>0 or Lw(520)>0), then
        RATIO = [max(Lw(443),0) + max(Lw(520),0)] / Lw(550)
        P = 5.56 * RATIO**(-2.252); else,
2. P = 46.24456 (saturated).
```

This algorithm does not involve a switching of equations that often results in a minimum for the pigment frequency distribution of the two-channel algorithm.

Finally, L2MULT and CLRWAT allow the user to change the calibration of bands 1 to 4. In the case of multiple scattering, only two options are provided, "Evans" and "User." The parameter is CORR. The CZCS suffered a severe calibration degradation or sensitivity loss which was erratic. Several algorithms were proposed by various investigators to correct for this and all are necessarily coupled to the Rayleigh scattering model they used. The reason for this coupling is that all techniques utilize either direct sea truth measurements or assume clear water radiances over the open ocean and the calibration is adjusted so as to match those values with the assumption that the Angstrom exponents in clear water regions are zero.

The Evans scheme (unpublished; used in the global CZCS processing) is the only one available which corresponds to the multiple scattering model of Gordon et al. (1988). The user should only try defining his own correction factors (FACTOR) when testing the sensitivity of the level-2 products or when he is trying to compare with sea truth observations. These calibration factors only multiply the calibration term as given in Gordon et al. (1983) and do not change the slope and intercept numbers in that term. The Evans calibration modifies the slopes and intercepts for each gain setting and includes a time-dependent correction factor as well. There must be a consistency between the calibration used in deriving the Angstrom exponents and that used in L2MULT.

The conversion of CZCS counts to total radiance (Lt), or calibration, uses the following general equation:

```
Lt(b) = [Counts(b)*SLOPE(b) + INTCP(b)] * FACTOR(b)
```

where SLOPE and INTCP are the equations slope and intercept, FACTOR is the correction factor, and b is a CZCS band (channel) number 1 to 4. For the "Evans" option, SLOPE, INTCP, and FACTOR are calculated as follows:

```
SLOPE(b)  = SLP(b,g) * MULTG(b,g) * SMULT(b) *
                [C(b,o) - (A(b,o)*Orbit)]
INTCP(b)  = INT(b,g) * MULTG(b,g) * IMULT(b)
FACTOR(b) = 1.0
```

Table 1. Values of time-independent "Evans" parameters.

parameter	gain	band 1	band 2	band 3	band 4
SLP	1	0.04452	0.03103	0.02467	0.01136
	2	0.03589	0.02493	0.02015	0.00897
	3	0.02968	0.02032	0.01643	0.00741
	4	0.02113	0.01486	0.01181	0.00535
INT	1	0.03963	0.05361	0.06992	0.01136
	2	0.03963	0.06361	0.06992	0.01136
	3	0.03963	0.06461	0.08292	0.01136
	4	0.03963	0.06361	0.07992	0.01136
MULTG	1	1.057	0.969	0.958	1.008
	2	1.060	0.970	0.947	1.020
	3	1.050	0.975	0.931	1.016
	4	1.059	0.960	0.934	1.010
SMULT	--	0.983	1.013	1.017	1.000
IMULT	--	1.0	1.0	1.146	1.0

where SLP and INT are the unmodified calibration slope and intercept, MULTG is a gain-dependent multiplier to SLP and INT, SMULT and IMULT are gain-independent multipliers to SLP and INT, and C and A are the intercept and slope of a time-dependent (orbit number) correction to SLP. The indices b and o represent the sensor gain and orbit. The values of SLP, INT, MULTG, SMULT, and IMULT are listed in Table 1, and those of C and A are listed in Table 2. For the "user" option, SLOPE and INTCP are obtained from Table 3 and FACTOR is as specified by the user.

Once the level-2 products have been created, gray level and geophysical values may be examined using programs such as READ, RLINE and HIST. For linearly scaled quantities, the slope and intercept are stored in the file header and are read by these programs in order to transform gray levels to geophysical values. READ allows the user to roam the image and examine values within a user-defined box or at individual points. RLINE allows the user to examine values along lines and HIST can be used to examine the frequency distributions.

Table 2. Values of time-dependent "Evans" parameters.

par.	orbit no.	band 1	band 2	band 3	band 4
A	<5001	-1.700E-05	-5.000E-06	-2.000E-06	0.0
	5001-6750	-6.000E-05	-5.000E-06	-2.000E-06	0.0
	6751-20000	-1.457E-05	-9.770E-06	-6.620E-06	0.0
	>20000	-1.700E-05	-6.000E-06	-5.000E-06	-5.000E-06
C	<5001	1.0	1.0	1.0	1.0
	5001-6750	0.785	1.0	1.0	1.0
	6751-20000	1.092	0.967	0.968	1.0
	>20000	1.0426	1.042	0.9995	0.9

Table 3. Slopes and intercepts for "user" option (from Gordon et al., 1983).

parameter	gain	band 1	band 2	band 3	band 4
SLOPE	1	0.04452	0.03103	0.02467	0.01136
	2	0.03598	0.02493	0.02015	0.00897
	3	0.02968	0.02032	0.01643	0.00741
	4	0.02113	0.01486	0.01181	0.00535
INTCP	1	0.03963	0.06361	0.07992	0.01136
	2	0.05276	0.08826	0.06247	0.03587
	3	0.02879	0.09752	0.06570	0.02963
	4	0.03359	0.05647	0.04723	0.01646

All image files are in a one byte per pixel, binary format with pixel values scaled from 0 to 255. In programs such as L2MULT, the user may decide the water radiance scaling by setting the radiance limits using the WATER parameter, but the defaults are 0 to 2.55 mw/(steradian-micron-cm^2). The water, aerosol and 443 Rayleigh radiances are all linear functions of gray level. The water radiance values are adjustable because some scenes may have features with radiances greater than 3.0. Aerosol radiance is scaled for values 0 to 2.55. Rayleigh radiance is scaled according to the minimum and maximum values for the scene and, therefore, its scaling varies from scene to scene.

For pigment concentrations (mg/m^3), PC-SEAPAK calculates the gray level values using the Univ. of Miami DSP system's scaling conversion:

GRAY = nint((log10(PIGMENT) + 1.4) / 0.012)

with gray scale limits of 2 through 245 (or 1.5 to 245.49 before rounding, corresponding to pigment values of 0.0415 to 35.15). ("Nint" is a function to round to the nearest integer.)

Acknowledgements: The tables used by PC-SEAPAK for the Rayleigh scattering computations were contributed by Howard R. Gordon, James W. Brown, and Robert H. Evans of the University of Miami. Values for Tables 1 and 2 presented here are from Robert H. Evans.

PROJECTING MULTIPLE IMAGES TO A COMMON MAP

The projection of images to a common map is often required in image analysis when studying a set of associated images. The study of a time series of images over a certain general region (Case 1) or the use of a number of images to form a single composite image covering a wider geographical area (Case 2) are two occasions where such projection is required. In both cases the images are projected to a common imaginary map and the images may or may not actually overlap with each other in geographical area. This section will describe how the projection program MAPIMG may be used to perform projection of multiple images for these two cases.

When the navigation information associated with an image is incorrect, the geocoordinates (as obtained by the program LATLON, for example) of image landmarks will also not be correct. Such images may be corrected using the program REGIST which will simply shift the gray level values relative to their pixel/line (TV) coordinates. (A monitor display of 512 pixels by 512 lines is assumed in this discussion.) This correction may be done independently of the use of MAPIMG which will map the corrected or uncorrected image using the same navigation information. Navigation data for CZCS scenes are usually accurate to within three pixels.

When using MAPIMG it is useful to think of the display (monitor) as a window or view area over an imaginary map of the world. A mapped image output by MAPIMG will normally have a portion or all of the area of the input image visible within this window. For Case 1, output images are often partially outside the window (lost) since each image in the series of images is likely to cover the earth area of interest to a different extent. Therefore the map of the world being considered is often much larger than the window for Case 1. For Case 2 however, output images are likely to be entirely within the window which will cover a large portion of, if not the entire, map. Case 2 output images can also be thought of as various pieces of a map puzzle (or mosaic) which may or may not be completely filled in.

"Output image" as used here refers to the input image as it would appear on the world map. The actual image created by MAPIMG, and contained in the file OUTFILE, is that of the window area of the map which may or may not include all of this "output image." If part of the output image is outside the window, it will be lost (i.e. not included in OUTFILE). Conversely, window pixels that are not within the output image boundary will be black (and stored in OUTFILE as such).

Controlling the Projection Characteristics of the Output

The MAPIMG input parameter PROJECTN determines the projection of the output image as well as of the imaginary map of the world of

which it is a part. Although the image is projected onto that map, the input parameters LL_1, LL_2, PIXEL, LINE, and DELTA_P allow the user to control where the window will be positioned over the map as well as the scale of the map. If defaulted, these parameters will be set such that the window will be directly over the output image area of the map and the map's scale will be such that this image will take up as much of the window as possible while remaining entirely within it. These default values are optimal for cases where images are being studied individually instead of as a group. Since these five parameters are used in conjunction with each other, if any one of them is defaulted, they will all be defaulted regardless of any values entered for some.

If no defaults are used, the PIXEL and LINE parameters refer to the TV coordinates of a point on the window whereas LL_1 refers to the geocoordinates (latitude and longitude) of a point on the world map. The points will be associated so that the window point overlays the map point. For a Case 1 study, the user may find it convenient to choose LL_1 to be in the center of the earth region of interest and assign it PIXEL/LINE values of 256/256, the center of the display (window). Alternatively, the user may wish that a certain landmark appear at a certain location on the display. In such a case, the landmark's geocoordinates would be entered for LL_1 and the desired display location specified by PIXEL/LINE. Similarly for Case 2, the geocoordinates of the geographical center of the desired composite image may be entered for LL_1 and 256/256 for PIXEL/LINE.

The use of parameters LL_2 and DELTA_P in conjunction with LL_1, PIXEL, and LINE controls the scale of the map and, hence, also controls how much of the mapped image appears on the display (i.e., within the window). LL_2 represents the geocoordinates of another point on the world map and DELTA_P represents the separation in pixels between that point and the PIXEL/LINE window location. A positive DELTA_P represents a horizontal separation, whereas a negative value represents a vertical separation.

Note that this second point need not be within the window and that the absolute value of DELTA_P may be larger than the display width or height (512 pixels or lines). For given parameters, a larger absolute DELTA_P will decrease the geographical area covered by the window (enlarge the map); a smaller absolute value will increase this area (contract the map). The direction of the second point relative to the first--that is, where they both fall on the world map--is determined solely by the projection.

Although DELTA_P represents the separation in either the horizontal or vertical direction (not the absolute separation), the points for LL_1 and LL_2 must be chosen such that they have both a horizontal AND vertical separation on the imaginary map. Therefore some a priori knowledge of where these points will fall on that map is required when choosing these parameter values.

A convenient way to determine values for LL_2 and DELTA_P is to use the geocoordinates of another landmark for LL_2 and enter the desired separation between LL_1 and LL_2 for DELTA_P. Another convenient way to determine these values is to determine the scale

for the map at LL_1. (In certain cases, depending on the projection, the scale will vary greatly even within the window area.) That is, the user decides how many display pixels (DELTA_P) should separate a longitudinal or latitudinal degree, minute, or second and assign LL_2 and DELTA_P accordingly. For example, if the scale at LL_1 is to be one latitudinal degree per 100 pixels and LL_1 is 10 degrees latitude and 38 degrees longitude, LL_2 would be 9 and 38 degrees and DELTA_P would be -100 (assuming that north is on top for this projection).

MAPIMG will prompt the user for dynamic parameters soon after it has been initiated. The requested parameters will depend on the projection that was specified.

It is important to understand that the default values for these dynamic parameters are calculated on the basis of the input image since the geographical characteristics for the map within the window have not been determined at this point. (These parameters are themselves used in the projection calculations and so, what the map would look like cannot be known at this point.) For example, if PROJECTN is 1 (UTM projection), the value for ZONE will be the number of the UTM zone in which is located the longitudinal midpoint of the input image. This ZONE value may be different from that desired for the common map onto which a series of images are being projection as described in the following paragraphs.

The dynamic parameter default values are meant to serve as a best guess for the map area that appears within the window area assuming that LL_1, LL_2, PIXEL, LINE, and DELTA_P are defaulted. Therefore it is up to the user to set these values so that they correspond to the desired map region within the window. For example, if the midpoint of this region is used for the zone in a UTM projection, this region will be at the center of a UTM-projected (imaginary) world map; if a point to the west of this region is used to determine this zone, the region will be on the right side of such a map.

When the map area within the window is large relative to the output image, as is often true for Case 2 studies, it is important to visualize this map area when deciding these dynamic parameter values. For instance, if an entire Van der Grinten (PROJECTN=19) world map will be within the window, the user may choose the central meridian such that the center of the map features the Americas, Europe/Africa, or the Atlantic or Pacific Oceans.

The input parameter ASPECT enables the user to modify the aspect ratio of the world map by stretching or contracting the horizontal and vertical aspects independently of each other. ASPECT is applied after the projection characteristics are determined by the program and so is independent of all other parameters. Since ASPECT may change the technical characteristics of a projection (a Van der Grinten projection, for example, will no longer be a Van der Grinten projection if ASPECT is not 1), a value other than 1 (the default) should only be used for special purposes such as to allow room on the display area for a caption, for example, or to permit a split screen display of more than one image.

Once the input parameters PROJCTN, LL_1, LL_2, PIXEL, LINE, DELTA_P, and ASPECT and the dynamic parameters have been established for one image, the same values must be used for the other input images in the set. This applies for both, Case 1 and 2, studies. These parameters determine the characteristics of the world map onto which the input images will be projected as well as the view area of the window.

Executing MAPIMG in Batch Mode

Because MAPIMG can only work on one input image file and it may require several minutes for each execution, a batch version has been designed to enable batch execution that allows up to 10 input image files, each with its own input parameter set, to be executed together. Two steps are required to execute this version. First, the program PARMPIMG is used to create or modify an input parameter file that can contain up to 10 independent input parameter sets. The second and main processing step is to use the program BHMAPIMG with the input parameter file generated by PARMPIMG to generate the mapped output images based on the input parameter sets contained in that file.

USING STATDIS TO GENERATE IMAGES FROM IMAGE DATA FILES

Some PC-SEAPAK programs create image "data" files instead of regular PC-SEAPAK image files. Regular files use a byte to store the value of each image pixel so that each pixel may have an integer value of 0 to 255. Data files, on the other hand, use a real number (four bytes) to represent each pixel. The precision for real data is much greater than for byte data: about one part per 8.4 million versus one part per 255. It is therefore very advantageous to perform image calculations using real pixel numbers instead of bytes values. For this reason, programs such as ADDF, MEANF, MULTF and LOGF generate data image files as output which may then be converted to regular image files using the program STATDIS. The image files generated by STATDIS may then be displayed directly on an MVP-AT frame buffer.

The optimal conversion (mapping) of the real data into values of 0 to 255 depends on the distribution of the data. The most straightforward mapping would be to assign the minimum to 0, the maximum to 255, and interpolate linearly for intermediate values. This is the mapping function used by STATDIS to display the initial image from an input linear data image. However, if the data are concentrated over a relatively small portion of their range, this mapping may result in a great loss of resolution for these data. Therefore a primary function of STATDIS is to allow the user to optimize a linear mapping by specifying a smaller data range to be mapped. STATDIS may also be used to assign the same mapping to multiple images of the same geophysical quantity since it is desirable that for such images the gray shades (or pseudocolors) represent the same magnitudes.

Note that since the pigment concentration-to-gray-level mapping is non-linear and is preset, STATDIS does not permit pigment image mappings to be varied.

Because of the larger storage requirements of data image files (four bytes instead of one per pixel), a user may choose to create such files only for a certain region of interest defined by a blotch. If so, that blotch must be used with STATDIS so that STATDIS can reconstruct the image area represented by the corresponding data image file.

The following describes the possible steps needed to create, for a given blotch area, an image resulting from the mathematical manipulation of other images in PC-SEAPAK:

1. Use IMAGE to display the image(s) of interest.
2. Use BLOTCH to define a blotch for the region of interest over the displayed image.
3. Use BPSAV to save the blotch displayed as a disk file.
4. Perform the desired mathematical calculations on the image file(s) for the defined blotch and generate the data image file. Programs such as ADDF, MEANF, MULTF, and LOGF generate such files and allow the user to specify a blotch file for the calculations.

5. Invoke STATDIS while the blotch defined in step 2 is displayed on the MVP-AT.
6. Use STATDIS to determine the image gray level mapping or save the image as a regular image file.

FORMATS OF IMAGE AND CONTROL POINT FILES

Standard PC-SEAPAK image files that can be dropped into MVT-AT frame buffers for display are simple flat files of 512 logical records, each of which is 512-bytes long. Each logical record corresponds to an image line when displayed and each byte corresponds to the pixels on that line. Additional 512-byte records containing header information may precede the image data. These image files normally contain one such header record. Files containing overlay graphics (also referred to as blotch files), are identical in structure except that they do not include any header records.

Users need not normally be concerned with the contents of an image file's header. However, a header's information may be examined using the program DMPHDR. The program MODHDR is provided for the rare occasion when a user may wish to change certain of this information. A complete list of the parameters retained in the header, their data types, and their locations, is presented in Table 1. Note that parameters may be for informational purpose only (i.e., they do not impact any program's calculations), may have more than one meaning depending on the context (i.e. the settings of flags and the program in question), or may be reserved for future use.

Except for images containing gridded data, unmapped images created by such programs as the ingest programs and DSPIMG are each associated with a control point file. Such files contain the navigation information for their corresponding images and are denoted by the extension ".CTL" in their filenames. This information consists essentially of a set of geocoordinates (latitude/ longitude pairs) corresponding to a set of display coordinates (pixel/line pairs). These display coordinates are also referred to as control points. From these data, the geocoordinates of all other image pixels can be interpolated for program calculations requiring earth locations. The format of control point files is explained in the following paragraph, although the user is cautioned that modifying such files may cause errors in programs using these data.

Control point files are ASCII files containing certain variables in the following logical records:

record	parameter	Fortran format
1	NCPP, NCPL, "1"	3I10
2	CPPIX	8I10/
3	CPLIN	8I10/
4	LATMIN, LATMAX, LONMIN, LONMAX, DATLIN	4F12.7,I10
5	CPLAT	8F12.7/
6	CPLON	8F12.7/

Table 1. Contents of PC-SEAPAK image file header.

Byte	Description	Format
1-4	area code	I*4
5-6	start year	I*2
7-8	start day	I*2
9-12	start msec	I*4
13-14	orbit number	I*2
15-34	spare	
35-36	gain	I*2
37-38	thresh	I*2
39-40	solar elevation (scene center)	I*2
41-42	solar azimuth (scene center)	I*2
43-44	roll (scene center)	I*2
45-46	pitch (scene center)	I*2
47-48	yaw (scene center)	I*2
49-52	gray-to-data slope	R*4
53-72	spare	
73-76	gray-to-data intercept	R*4
77-96	spare	
97-98	starting pixel of tape ingesting	I*2
99-100	starting line of tape ingesting	I*2
101-102	ending pixel of tape ingesting	I*2
103-104	total lines on tape ingesting	I*2
105-106	pixel reduction factor	I*2
107-108	line reduction factor	I*2
109-110	spare (must be 0)	
111-112	tilt angle	I*2
113-128	spare	
129-132	minimum latitude in the image	R*4
133-136	maximum latitude in the image	R*4
137-140	minimum longitude in the image	R*4
141-144	maximum longitude in the image	R*4
145-146	control points per image line	I*2
147-148	control points per pixel column	I*2
149-164	image corner latitudes	4(R*4)
165-180	image corner longitudes	4(R*4)
181-184	increment in msec	I*4
185-200	four epsilons	4(R*4)
201-236	control point file name	A*36
237-256	five circle parameters	5(R*4)
257-258	display offset	I*2
259-260	stamp for derived images	I*2
261-262	spare	I*2
263-264	normalized water radiance flag	I*2
265-280	spare	
281-282	MAPIMG projection index	I*2
283-284	UTM or SPC zone	I*2
285-344	15 projection parameters	15(R*4)
345-346	spare	

Table 1. Contents of PC-SEAPAK image file header. (Cont'd)

Byte	Description	Format
347-348	data source	C*2
	C1 - CZCS level-1 image	
	C2 - CZCS level-2 image	
	A6,A7,A8,A9,AA,AB - AVHRR-6,7,8,9,10,11	
	M2 - U. of Miami DSP	
	G - gridded image	
349-350	data type	C*2
	L1 - CZCS or AVHRR level-1	
	TR - total radiance	
	PI - pigment	
	SC - CZCS SST	
	SA - AVHRR SST	
	WR - water radiance	
	RA - Rayleigh radiance	
	AT - AVHRR thermal bands	
	DA - diffuse attenuation	
351-352	band number	I*2
353-354	starting pixel of image	I*2
355-356	ending pixel of image	I*2
357-358	starting line of image	I*2
359-360	ending line of image	I*2
361-364	spare	
365-366	MAPIMG projection index	I*2
367-368	UTM or SPC zone	I*2
369-488	15 projection parameters	15(R*8)
489-512	spare	

Records 5 and 6 are repeated NCPL times, one pair for each CPLIN value. The definitions of these variables are:

- NCPP: Number of control points per image line (control point pixels); 1 to 100; usually 26.
- NCPL: Number of control points per pixel column (control point lines); 1 to 100; usually 26.
- "1": Flag indicating that it is a new format control point file, described herein; the new format was adopted in April 1988 to allow more flexibility in mapping images to different projections.
- CPPIX: Unmapped pixel indices of control points; run from small to large (left to right on the image display); NCPP values ranging from 1 to 512.
- CPLIN: Unmapped line indices of control points; run from small to large (top to bottom on the image display); NCPL values ranging from 1 to 512.
- LATMIN: Southernmost CPLAT value.
- LATMAX: Northernmost CPLAT value.
- LONMIN: Westernmost CPLON value.

LONMAX: Easternmost CPLON value.

DATLIN: Equals -1 if 180 deg. longitude crosses the control point field; else, equals 0; may be used as a logical variable.

CPLAT: Latitudes (+-90 decimal degrees) of control points; each CPLAT record contains NCPP values, one for each CPPIX value; there are NCPL CPLAT records, one for each CPLIN value; read in as a two dimensional array where the first index represents the horizontal (pixel) display direction and the second represents the vertical (line) direction; e.g., CPLAT(3,4), the third value of the fourth CPLAT record, is the latitude for the pixel CPPIX(3) on line CPLIN(4); the order of lines run N to S and the order of pixels run W to E.

CPLON: Longitudes (+-180 decimal degrees) of control points; each CPLON record contains NCPP values, one for each CPPIX value; there are NCPL CPLON records, one for each CPLIN value; read in as a two dimensional array where the first index represents the horizontal (pixel) display direction and the second represents the vertical (line) direction; e.g., CPLON(3,4), the third value of the fourth CPLON record, is the longitude for the pixel CPPIX(3) on line CPLIN(4); the order of lines run N to S and the order of pixels run W to E.

In PC-SEAPAK, level-3 images refer to those created by the projection program MAPIMG (and its alternate versions, BHMAPIMG, WTKMP, and BHWTKMP). Level-3 images contain the values of their projection equation parameters in their headers and are not associated with control point files.

PC-SEAPAK MENU TREE

DATE: 10/28/91

MENU/PROGRAM	DESCRIPTION
1. INGEST - CZCS and AVHRR ingestion	
A. CZCSIN - CZCS ingestion	
1. TPCZCS	Level-1 tape ingest
2. DKCZCS	Level-1 disk ingest
3. TP2DSK	Full-scene level-1 tape ingest
4. DSK2DSK	Full-scene level-1 disk ingest
5. WINDOW	Image from full-scene file
B. AVHRRIN - AVHRR ingestion	
1. TPAVHRR	NOAA-format level-1b tape ingest
2. DKAVHRR	NOAA-format level-1b disk ingest
3. TPSDRPS	NORDA-format level-1b tape ingest
4. DKSDRPS	NORDA-format level-1b disk ingest
5. SOLARZ	Solar zenith correction
2. CZCSL2 - CZCS level-2 processing	
A. L2BOX	Localized level-2 analysis
B. FLAGLC	Land/cloud flag determination
C. THRES	Threshold determination
D. ATMOS - Atmospheric correction and determination	
1. CLRWAT	Angstrom determination, auto mode
2. SCREEN	Valid clear water pixel display
3. BXCLRWAT	Angstrom determination, box mode
4. OZONE	Ozone/optical thickness values
5. ANGST	Interactive Angstrom determination
E. L2PROD - CZCS level-2 product generation	
1. L2MULT	Level-2 product generation
2. BHL2MULT	Batch mode of L2MULT
3. PARL2MU	L2MULT parameter file generation
4. WTKLM	L2MULT for WEITEK coprocessor
5. BHWTKLM	BHL2MULT for WEITEK coprocessor
6. L2CON	Pigment scaling conversion
F. RING	Mask out sensor ringing
3. IMAGING - MVP-AT frame buffer programs	
A. INITIAL - MVP-AT initialization	
1. INIT	MVP-AT initialization
2. CLR	Clear frame buffers
3. SPKSETUP	Configuration setup
4. GPCOLOR	Graphics palette color assignment
5. BPCOLOR	LUT palette color assignment
6. BP1COLOR	Individual palette color assignment

3. IMAGING - MVP-AT frame buffer programs (continued)

B. FRMBUF - Frame buffer operations

1. IMAGE . . . . Display image
2. IMAGSAV . . . . Save image into file
3. IMGXRT . . . . Display non-SEAPAK image
4. SELECT . . . . Select frame buffer
5. LOOP . . . . Loop frame buffer images
6. DSKLOOP . . . . Loop disk file images
7. IMGLST . . . . List frame buffer image names
8. ZOOM . . . . Zoom and roam an image
9. IMGEDIT . . . . Cut and shift an image

C. OVERLAYS - Overlay graphics manipulation

1. ANNOTATE . . . . Annotation
2. BLOTCH . . . . Area-of-interest definition
3. CONTOUR . . . . Gray-level contouring
4. GPONOFF . . . . Turn graphics on/off
5. GPCLR . . . . Clear graphics
6. BPSAV . . . . Save graphics into file
7. BPLOAD . . . . Load graphics from file
8. REGION . . . . Area-of-interest border overlay
9. TRACK . . . . Aircraft/ship track data analysis

D. LUTCOLOR - Look-up table and color operations

1. TABLOAD . . . . Load LUT into frame buffer
2. TABSAV . . . . Save LUT into disk file
3. STRETCH . . . . Linear contrast stretch
4. PLI . . . . Piecewise linear stretch
5. LUTMOD . . . . Gray-level range rescale
6. RESCALE . . . . Image rescaling
7. PAINT . . . . False color palette development
8. COLBAR . . . . Generate color bar

E. MOSAIC - Mosaic programs

1. MOSAIC . . . . Mosaic image creation
2. RGBDIS . . . . True color image display

4. IMGFILE - Image file information

A. DATA - Image file data retrieval

1. READ . . . . Data extraction (box/point)
2. RLINE . . . . Data extraction (line)
3. ASCIMG . . . . 2-D array from image

B. HEADER - Image file header information

1. DMPHDR . . . . Display/print file header
2. MODHDR . . . . Change header information
3. ADDHDR . . . . Add header block to image file

C. MERGE . . . . Merge CZCS/AVHRR images

D. ALIAS . . . . Assign aliases to file names

E. HILOW . . . . Min/max image generation

5. <u>GEOGRAPH</u> - Geographic programs

   A. <u>PROJECTN</u> - Image map projection programs

      1. MAPIMG . . . . Image map projection
      2. BHMAPIMG . . . . Batch mode of MAPIMG
      3. PARMPIMG . . . . MAPIMG parameter file generation
      4. WTKMP . . . . MAPIMG for WEITEK coprocessor
      5. BHWTKMP . . . . BHMAPIMG for WEITEK coprocessor

   B. REGIST . . . . Image translation

   C. GRID . . . . Latitude/longitude grid overlay

   D. LATLON . . . . Latitude/longitude determination

   E. COAST . . . . Coastline/geographic features overlay

   F. BATHYIMG . . . . Bathymetry image file generation

6. <u>MATH</u> - Mathematical programs

   A. <u>HARDFCT</u> - Hardware image functions

      1. IMGFCT . . . . Functions for image and constant
      2. IMG2FCT . . . . Functions for two images
      3. FILTER . . . . Filtering and edge detection
      4. CONVOLVE . . . . Image convolution

   B. <u>SOFTFCT</u> - Software image functions

      1. ADDF . . . . Add./subt./exp. of image file(s)
      2. MEANF . . . . Averaging of image files
      3. LOGF . . . . Logarithm of an image file
      4. MULTF . . . . Mult./div./exp. of two image files
      5. STATDIS . . . . Image from real-valued data file
      6. DIFFI . . . . Difference of two image files
      7. DERIV . . . . Derivatives of an image
      8. IMATCH . . . . Modify image based on another image

   C. <u>STAT1</u> - Statistical analyses menu 1

      1. HIST . . . . Frequency distribution
      2. BHHIST . . . . Batch mode of HIST
      3. MV . . . . Mean/variance image generation
      4. FILLA . . . . Image fill/filter, auto mode
      5. FILLM . . . . Image fill, manual mode
      6. TSERIES . . . . Time series statistics plot
      7. MEM . . . . Maximum entropy method

   D. <u>STAT2</u> - Statistical analyses menu 2

      1. SCATT . . . . Scattergram plot
      2. CORCO . . . . Correlation coefficient
      3. AUTOCORR . . . . Autocorrelation vs. lags
      4. XCORR . . . . Cross correlation vs. lags
      5. VARIOG . . . . Semivariance vs. lags
      6. EOF . . . . Empirical orthogonal function analysis
      7. EOFPLOT . . . . Plot/MEM of EOF principal components

7. UTIL - Utility programs
   A. HARDCOPY - Image hard-copy programs
      1. PJTCOL . . . . PaintJet color selection
      2. IMGPRINT . . . . PaintJet image hard copy
      3. PSIMAGE . . . . Postscript image hard copy
   B. MIAMI - Univ. of Miami file format support
      1. PSTIMG . . . . Image from PST file
      2. DSPIMG . . . . Image from DSP file
      3. NODSST . . . . Image from NODS MCSST file
      4. PSTXRT . . . . Image from PSTIMG image
   C. VAXTOPC - VAX-to-PC format conversion
      1. HDRCVT . . . . Header block conversion
      2. BLOCVT . . . . Graphics file conversion
      3. LUTCVT . . . . LUT file conversion

PROGRAM NAME: ADDF
DATE: 10/28/91
MENU: SOFTFCT

DESCRIPTION: This program may be used to add or subtract several disk image files, pixel by pixel, according to the following general equation:

OUT = C + sum[W(n)*I(n)**E(n)] for n = 1 to NUM

where OUT is the output data file designated by the parameter OFIL, C corresponds to the constant CONST, W are the weights WEIGHT, I are the image data from the files IN_FIL, E are the exponents EXPONENT, and NUM is the number of IN_FIL files. The image region of interest may be specified by GPAL and BFIL.

The calculation results are stored as real-valued data in OFIL in order to retain maximum accuracy. OFIL may be used subsequently as input to the program STATDIS in order to generate its image, optimize its gray scale, and save it as a disk image file. For a given pixel, if any I(n) value falls outside the RANGE values, or if an arithmetic error occurs during summation, OUT for that pixel will be flagged as "invalid" and subsequently assigned a value that is specified in STATDIS.

PARAMETERS:

(1) **IN_FIL** is the array of the input image file names to be processed. Up to 12 files may be entered at once. All files should contain one header block (512 bytes) followed by 512 blocks of image data.

(2) **WEIGHT** is the array of the weighing factors for IN_FIL. A number must be entered for each IN_FIL to be summed. Each number will be used as a multiplicative factor for the pixel values of its corresponding image (raised to the EXPONENT power) during summation. To illustrate the use, consider the following examples:
   1) for simple summation, set CONST=0, WEIGHT(n)=1, and EXPONENT(n)=1;
   2) to raise a single image to the 3rd power, set CONST=0, WEIGHT(1)=1, and EXPONENT(1)=3;
   3) to subtract image 2 from image 1, set CONST=0, WEIGHT(1)=EXPONENT(1)=EXPONENT(2)=1, and WEIGHT(2)=-1.

(3) **EXPONENT** is the array of the exponents for IN_FIL. A number must be entered for each IN_FIL to be summed. Each number will be used as the power by which to raise the pixel values of its corresponding image during summation. Note that EXPONENT not equal to one will affect the units of their respective terms. It is the user's responsibility to ensure that the final units of terms are consistent. Arithmetic errors may occur during summation if inappropriate EXPONENT values are used. For example, errors will occur if EXPONENT is too large or too small, or if negative EXPONENT is used

with zero or negative input image pixel values. Output data values of pixels for which arithmetic errors have occurred will be flagged as "invalid" and may be assigned any desired value when using the program STATDIS. (See the documentation for the program STATDIS dealing with the parameter INVAL for further information). Such pixels cannot be distinguished from those flagged as "invalid" because of range restrictions which are described later. ADDF will display the number of pixels with such errors, if any have occurred, at the end of its processing. With the use of an appropriate blotch or values for RANGE, these pixels may be excluded from the calculations. However, these arithmetic errors may indicate that your values for EXPONENT and other input parameters are incorrect and should be changed.

(4) **OFIL** is the name for the "data" file output to the disk. This file is composed of floating point numbers for higher accuracy. OFIL may be used as input to the program STATDIS in order to generate its image, optimize its gray scale, and save it as a PC-SEAPAK image file. Note, however, that the same blotch specification used in ADDF will be needed by STATDIS (i.e., the same blotch must be used unless GPAL=0). "Data" files such as OFIL cannot be dropped directly into the image display unit as images or used as input to this program. STATDIS must be used to generate and save image files from "data" files. In this way, you can interactively obtain, using STATDIS, an optimum gray scale for the image file corresponding to the range or subrange of data values in the "data" file. By convention, "data" file names end with the extension ".DAT" whereas image file names end with ".IMG". Note that the disk space required by a "data" file is proportional to the blotch area and may be much more than that required by an image file which is always 513 blocks. For a full image (GPAL=0, the equivalent of a full-image blotch), a "data" file will require 2049 blocks or about four times the space of an image file; for a blotch covering less than a quarter of the image, however, the "data" file will be smaller than an image file.

(5) **MODE** is a flag which indicates whether the pixel values of the IN_FIL image(s) represent data (such as temperature or radiance) that are linearly related to gray levels, or pigment concentrations which are non-linear. A "1" should be entered for linear data and a "2" for pigment data.

(6) **RANGE** defines the range of IN_FIL pixel values to use for the summation. The user should enter two values in the input data units. For a given pixel location, if a value for any IN_FIL falls outside the RANGE values, the corresponding pixel in OFIL will be flagged as "invalid." These "invalid" pixels may be assigned any value when using STATDIS to generate the image from OFIL. Again, the RANGE values must conform to the units of the IN_FIL image(s) as specified by MODE and FACTOR (i.e. pigment concentration or units linearly proportional to gray levels). For example, to exclude only land and cloud pixels,

the RANGE values should be 1.0 and 254.0 (the default values) for gray levels (MODE=1 and FACTOR=1) or 0.0409 and 44.46 for pigment concentrations (MODE=2).

(7) **CONST** is a constant (in output data units) which is to be added to the summation. The user should enter a real number whose units match those of the other terms.

(8) **FACTOR** is a linear scale factor used only if MODE=1, i.e. when a linear data-to-gray scale mapping function for the IN_FIL image(s) is used. If greater than zero, it will represent the factor by which to divide the gray values of IN_FIL pixels in order to convert them into actual data values; if zero or less, the slope and intercept for this mapping function will be obtained from each file header of the IN_FIL disk image files. In order to retain the gray values, enter 1 (the default value); for sea surface temperature (SST), enter 8; for water radiance data, enter 85; for aerosol radiance data, enter 100.

(9) **GPAL** is the graphics palette which defines the blotch area(s) of interest and is in the range -7 to 7. If the number entered is positive, pixels within the blotch will be considered. If the number is negative, pixels outside the blotch will be considered. Only blotches defined by this graphics palette (the absolute value of GPAL) of the blotch file BFIL will be used. If "0" is entered, the entire image area (512 x 512) will be used and BFIL will be ignored.

(10) **BFIL** is the name of the blotch file which defines the image area(s) of interest unless GPAL= 0. Only blotches defined by the graphics palette corresponding to GPAL will be used. Blotches may be drawn and saved as files using the programs BLOTCH and BPSAV.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: ADDHDR
DATE: 10/28/91
MENU: HEADER

DESCRIPTION: This program allows the user to add a header block (512 bytes) at the beginning of an image file or a group of image files. The header block to be added may be a blank or a header from another image file.

PARAMETERS:

(1) **IFIL** is the file name that specifies an image file or a group of image files to be inserted with a header block (512 bytes) at the beginning of each image file. The header block to be added is specified by the parameter HFIL. The wild card characters "*" and "?" may be used to specify a group of files: "?" to replace a single character and "*" to replace multiple characters in the IFIL name.

(2) **HFIL** is the file name whose header block (the first 512 bytes) is to be added into the file(s) specified in IFIL. If this parameter is blank, a blank header block will be used.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: ALIAS
DATE: 10/28/91
MENU: IMGFILE

DESCRIPTION: This program allows the user to assign file name(s) to alias name(s) so that alias names may be used for file specifications in the program IMAGE. Note that to use an alias name, the user must enter an apostrophe, " ' ", before the alias name. A list of all alias names is kept in the ASCII file ALIAS.PAR in the SEAPAK directory. This file may be printed or edited using regular DOS commands. If the parameter FILNAM is blank, this program can also be used to list the file name(s) assigned to the alias name(s) which match the specification of the parameter ALIAS.

PARAMETERS:

(1) **ALIAS** is the alias name. If FILNAM is blank, the program will list all alias names that match the specification of ALIAS and the file names assigned to them. The wild card character "*" may be used anywhere in the ALIAS character string when FILNAM is blank. If FILNAM is not blank and does not contain "*", ALIAS will be recorded as the alias name of FILNAM. If there is "*" in FILNAM, the characters that replace the "*" in each file name found to match the FILNAM specification will be appended to ALIAS to form the corresponding alias names.

(2) **FILNAM** is the file name to be assigned to the alias name specified in ALIAS. If FILNAM does not contain the wild card character "*", the program will just assign the file name FILNAM to the alias name ALIAS. If there is "*" in FILNAM, the characters that replace the "*" in each file name found to match the FILNAM specification will be appended to ALIAS to form the corresponding alias names. (The "*" must be the last character prior to the period of the file name extension.) For example, if there are four files TEST1.IMG, TEST2.IMG, TEST3.IMG, TEST4.CTL in the current directory and ALIAS is "XX" and FILNAM is "TEST*", these files will be assigned to the alias names XX1, XX2, XX3, and XX4. However, if FILNAM is "TEST*.IMG", then only the first three files (with extensions ".IMG") will be assigned to the alias names XX1, XX2, and XX3. If FILNAM is blank, the program will list all alias names that match the specification of ALIAS and the file names assigned to them.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: ANGST
DATE: 10/28/91
MENU: ATMOS

DESCRIPTION: ANGST provides an alternative method to CLRWAT for estimating the Angstrom exponents required for removing aerosol radiance from a CZCS level 1 scene. The technique is based on Arnone and LaViolette (1984) and allows the user to interactively select the Angstrom exponent for the aerosol correction, correct the image and read the resulting subsurface water radiance values. The atmospheric correction algorithm is based on Gordon et al. (1988).

The user can visually correct a scene which may not have clear water or has distinct haze bands. When the aerosols have sharp structure, the Angstrom exponent can be increased until the structure disappears. The haze will appear as bright areas in the undercorrected scene, but if the scene is overcorrected, the structure will become darker than adjacent clear areas.

The correction is performed on each of channels 1 through 3 independently rather than simultaneously as in CLRWAT. Thus, the program requires that the level 1 radiance image files from any one of channels 1, 2 or 3 and channel 4. Channel 4 image file is required since the aerosol correction is based on the residual radiance in channel 4 after it has had the Rayleigh component removed. The channel 5 image file may be used, but not required, for the reference of land/clouds.

Once the program is initiated, it removes the Rayleigh radiance from the level 1 data. After this is accomplished, the user may change the Angstrom exponent by a horizontal translation of the cursor using the mouse or cursor keys. Function key F2 is used when a correction is to be made.

One approach of using ANGST is to follow some of the procedures in the 'clear-water' radiance algorithm which is applied in CLRWAT. First, correct channels 2 and 3 remembering that the absolute magnitude of the Angstrom exponent for channel 2 is usually greater than or equal to that for channel 3, and then average the Angstrom exponents in channels 2 and 3 and apply that value to channel 1. Continental haze has Angstrom exponents greater than or equal to zero; marine haze has Angstrom exponents of about zero; and dust has Angstrom exponents less than or equal to zero.

PARAMETERS:

(1) **IFIL1** is the file name of a level 1 radiance scene to be corrected. CZCS level 1 band 1, 2, or 3 image file should be used. The IFIL1 and IFIL2 have to be from the same scene.

(2) **IFIL2** is the CZCS level 1 band 4 image file. The IFIL1 and IFIL2 have to be from the same scene.

(3) **IFIL3** is the land/clouds reference file. The CZCS level 1 band 5 image file should be used.

(4) **LANCLD** is the channel 5 threshold in gray level value used to identify land and clouds. All pixels with values exceeding this value are assigned a value of 255.

(5) **CORR** is the index of the correction method to use for calculating total radiances:
   1: Use factors and method (time and gain dependent) of R. Evans (Univ. of Miami).
   2: Use correction factors specified by FACTOR.

(6) **ILTOPT** specifies the ILT option: If "1", ephemeris data from the ILT record of the level-1 scene will be used. If "0", much of these data will be obtained from the documentation record or calculated by SEAPAK based on the location and time at the start of the scene.

(7) **GPAL** is the graphics palette used to mark the location of cursor.

(8) **ANGS_RNG** are minimum and maximum Angstrom exponent values. The values set the range of values that can be applied for the aerosol correction and correspond to the extreme right side of the screen and extreme left side of the screen, respectively, when moving the cursor horizontally to select an Angstrom exponent.

(9) **OZONE** are optical thicknesses (in meters) for bands 1 to 4. If the value "-999" is entered, the values used will be from the PC-TOMS database for the day of the input CZCS scene and for the point nearest to the image center. If the PC-TOMS data point is missing or an error occurs accessing the data, a message to that effect will be displayed on the terminal along with the default values. If defaults are used, the values will be 0.00106, 0.0144, 0.0279, and 0.0125. These thicknesses are the products of the absorption coefficients (3.4E-6, 46E-6, 89E-6, and 40E-6) used at the Univ. of Miami and an average amount of 313 Dobson units of ozone.

(10) **FACTOR** are the correction factors to use for calculating total radiances of bands 1 to 4, respectively. These will be used only when CORR=2.

DYNAMIC PARAMETERS:

I. Parameter for outputting subsurface water radiance image into a file.

(1) **OFIL** is the output file name to be used to save the subsurface water radiance image.

FUNCTION KEY DEFINITIONS:

ESC: Exits the program.

F1: Displays current Angstrom exponent value. Any single horizontal movement will increase or decrease current value by (max(ANGS_RNG) - min(ANGS_RNG)) / 512.

F2: Displays the subsurface water radiance image by removing the aerosol radiance, which is calculated with current Angstrom exponent value, from the Rayleigh corrected level 1 band 1, 2, or 3 image.

F3: Displays the subsurface water radiance values and the mean and standard deviation inside current cursor box.

F4: Outputs the solar and the spacecraft zenith and azimuth angles used in the Rayleigh scattering computations. Additional information output are the subsurface "clear water" radiances that correspond to the solar zenith angle at the cursor location.

F5: Changes the size of the box cursor in a loop of sizes 1x1, 3x3, 5x5, 7x7, 9x9, 11x11, 13x13, and 15x15.

F6: Marks the cursor at current location.

F7: Allows the user to enhance the contrast of the corrected scene in order to determine more clearly the structure in the scene.

F8: Displays the cursor location (center of the cursor box) in pixel/line as well as in latitude/longitude coordinates.

F9: Displays the next image frame buffer.

F10: Allows the user to save current subsurface water radiance image into a file. Dynamic parameter OFIL will be requested.

ALT F1: Toggles the function key menu display on/off.

ALT F9: Allows the user to process another band. The parameter IFIL1 will be requested.

ALT F10: Allows the user to change the calibration parameters CORR and FACTOR. Once these are entered the sequence repeats by recomputing the total radiance and subtracting the Rayleigh radiance.

MOUSE LEFT BUTTON - Same as function key F1.

MOUSE RIGHT BUTTON - Toggles the function key menu display on/off.

PROGRAM NAME: ANNOTATE
DATE: 10/28/91
MENU: OVERLAYS

DESCRIPTION: The program ANNOTATE writes character/symbol overlays on the current displayed frame buffer. There are two text modes available, dot text and stroke text, each of which has different font selections available (see Appendix). The user can select the text mode and the font, change the size of the characters, the colors of the characters, the color of character's background (dot text) or fill (stroke text), the orientation of the text (dot text supports only 0, 90, 180, and 270 degree orientations). The user can write characters directly through keyboard or save the character strings into a buffer and write out that buffer at any time to any cursor position.

PARAMETERS:
None.

DYNAMIC PARAMETERS:
For changing parameters of dot text (press F1)

(1) **DOT_H** specifies the height in pixels of the dot text, it should always be a multiple of 8.
(2) **DOT_W** specifies the width in pixels of the dot text, it should always be a multiple of 8.
(3) **DOT_ANG** specifies the direction in degrees of the dot text to be displayed. Only 0, 90, 180 and 270 are valid inputs.
(4) **G_PAL** specifies the overlay color palette to be used for the dot text, only 0 to 7 are valid inputs. To change the color of the palette(s), use program BPCOLOR or BP1COLOR.
(5) **GB_PAL** specifies the overlay color palette to be used for the background of the dot text, only 0 to 7 are valid inputs. To change the color of the palettes(s), use program BPCOLOR or BP1COLOR.
(6) **POS_MODE** specifies how the annotation stored in the text buffer will be displayed relative to the cursor position.
    1. the cursor location is the start of the annotation.
    2. the cursor location is the end of the annotation.
    3. the cursor location is the center of the annotation.
(7) **DOT_FONT** specifies the font file selection for dot text. There are seven fonts available.
    1. HALO 88 default   2. HALO001.FNT   3. HALO002.FNT
    4. HALO010.FNT   5. HALO011.FNT   6. HALO012.FNT
    7. HALO013.FNT
    For font definitions, check the appendix in this guide.
(8) **TXT_BUF** specifies the text strings up to 80 characters to be saved in the buffer which can be displayed as the annotation whenever the function key F4 is pressed.

For changing parameters of stroke text (press F2)

(1) **STRK_FONT** specifies the font file selection for stroke text. There are 19 available fonts.

1. HALO102.FNT	2. HALO103.FNT	3. HALO104.FNT
4. HALO105.FNT	5. HALO106.FNT	6. HALO107.FNT
7. HALO108.FNT	8. HALO109.FNT	9. HALO111.FNT
10. HALO115.FNT	11. HALO201.FNT	12. HALO202.FNT
13. HALO203.FNT	14. HALO204.FNT	15. HALO205.FNT
16. HALO206.FNT	17. HALO207.FNT	18. HALO208.FNT
19. HALO209.FNT		

For font definitions, check the appendix in this guide.

(2) **STRK_H** specifies the height in pixels of the stroke text.

(3) **STRK_ANG** specifies the direction in degrees of the stroke text to be displayed.

(4) **STRK_ASP** specifies the aspect ratio of the stroke text.

(5) **G_PAL** specifies the overlay color palette to be used for the stroke text. Only 0 to 7 are valid inputs. To change the color of the palette(s), use program BPCOLOR or BP1COLOR.

(6) **FILL_PAL** specifies the overlay color palette to be used for filling the stroke text, only 0 to 7 are valid inputs. To change the color of the palette(s), use program BPCOLOR or BP1COLOR.

(7) **POS_MODE** specifies how the annotation stored in the text buffer will be displayed relative to the cursor position.
1. the cursor location is the start of the annotation.
2. the cursor location is the end of the annotation.
3. the cursor location is the center of the annotation.

(8) **TXT_BUF** specifies the text string (up to 80 characters) to be saved in the buffer which can be displayed as the annotation whenever the function key F4 is pressed.

FUNCTION KEY DEFINITIONS:

ESC: Exits the program.

F1: Allows one to change the current setups for the dot text.

F2: Allows one to change the current setups for the stroke text.

F3: Toggles text mode from dot text to stroke text.

F4: Causes the annotation stored in the stoke text buffer or the dot text buffer to be displayed depending on the current text mode and its status.

F5: Displays the next image frame buffer.

F6: Displays the pixel/line coordinates of the cursor position.

ALT F1: Toggles the function key menu display on/off.

MOUSE RIGHT BUTTON - Same as ALT F1

PROGRAM NAME: ASCIMG
DATE: 10/28/91
MENU: DATA

DESCRIPTION: ASCIMG is a program which can be used to generate an ASCII file of an image. Up to five PC-SEAPAK images or any rectangular subscene from them can be converted at one time to ASCII flat-file equivalents. This program not only allows one to select a portion of the scene for conversion to ASCII files but also allows one to subsample the scene or flip it around from left to right or top to bottom. Also, the format of the ASCII output file records may be specified by the user.

PARAMETERS:

(1) **IN_FILE1-IN_FILE5** contain the list of input image files one wants to convert to ASCII files. These files when converted will correspond to the output image files (OU_FILE1-OU_FILE5) and will all have been processed in the same manner.

(2) **OU_FILE1-OU_FILE5** contain the ASCII equivalents of the respective PC-SEAPAK image files listed in (IN_FILE1-IN_FILE5). Note that all these files will be generated using the same values for all the other input parameters. These files will not contain any header records. The format is specified by the user. See the parameter FORMAT for a detailed description of their record formats.

(3) **ST_PIX** defines the location of the start pixel for the rectagular area of the image(s) for which ASCII file(s) will be generated. ST_PIX/ST_LIN and END_PX/END_LN define the full rectangle. The value one inputs for ST_PIX depends on the location of the origin which in turn is determined by the parameter P_DIR and TOT_PX. In other words, if P_DIR=1, pixel positions are enumerated from left to right and ST_PIX is the pixel position of a left corner. If P_DIR=2, pixel positions are enumerated from right to left, and ST_PIX is the pixel position of the right corner of the rectangle. This means that the number entered for ST_PIX is determined by counting pixels from the right starting at the origin (which is defined by TOT_PX), e.g. the rightmost pixel (the origin) is 1, the next is 2, etc. Note that all this also means that ST_PIX can never be greater than END_PIX.

(4) **ST_LIN** is the location of the start line for the rectagular area of the image(s) for which ASCII file(s) will be generated. As mentioned earlier, ST_PIX/ST_LIN and END_PX/END_LN define the full rectangle. As for ST_PIX, the value one inputs for ST_LIN depends on the location of the origin which in turn is determined by the parameter L_DIR and TOT_LN. In other words, if L_DIR=1, line positions are enumerated from top to bottom; ST_LIN is the line position of a top corner. If L_DIR=2, line positions are enumerated from bottom to top and ST_LIN is the line position of a bottom corner of the rectangular box. In a similar fashion to ST_PIX, the number

entered for ST_LIN is determined by counting lines from the bottom starting at the origin (which is defined by TOT_LN), e.g. the bottommost line (the origin) is 1, the next is 2, etc. It is evident that this means that ST_LIN cannot be greater than TOT_LN. From all of this it is also clear that ST_LIN cannot be greater than END_LIN.

(5) **END_PX** is the location of the end pixel and helps to define one corner of the rectangular area of the image(s) for which ASCII file(s) will be generated. END_PX cannot be less than ST_PIX since the origin for counting the pixels starts from the ST_PIX direction. In other words, if P_DIR=1, pixel positions are enumerated from left to right and END_PX is the pixel position of a right corner. If P_DIR=2, pixel positions are enumerated from right to left and END_PX is the pixel position of a left corner. END_PX cannot be greater than TOT_PX.

(6) **END_LN** is the location of the end line and helps to define one corner of the rectangular area of the image(s) for which ASCII file(s) will be generated. Again due to the location of the origin, END_LN cannot be less than ST_LIN. Also, if L_DIR=1, line positions are enumerated from top to bottom and END_LN is the line position of a bottom corner. If L_DIR=2, line positions are enumerated from bottom to top and END_LN is the line position of a top corner. END_LN cannot be greater than TOT_LN.

(7) **P_SUBS** is the parameter which specifies the pixel subsampling rate. For example, if P_SUBS=2, only every other pixel from ST_PIX to END_PX will be processed and the resulting ASCII image (OU_FILEs) will be reduced by a factor of two in the pixel (horizontal) direction. Note that the last pixel processed for each line will be

    ST_PIX + (N-1)*P_SUBS

where N is the number of pixels processed per line or

    Integer((END_PX-ST_PIX + P_SUBS)/P_SUBS).

(8) **L_SUBS** defines the line subsampling rate. For example, if L_SUBS=2, only every other line from ST_LIN to END_LN will be processed and the resulting OU_FILEs will represent images reduced by a factor of two in the line (vertical) direction. Note that the last line processed for each file will be

    ST_LIN + (N-1)*L_SUBS

where N is the number of lines processed per image or

    Integer ((END_LN-ST_LIN + L_SUBS)/L_SUBS).

(9) **MODE** specifies whether the image data (IN_FILEs) are linearly related to gray scales or are pigment concentrations. Enter 1 (the default value) if the pixel values of the displayed image represent data such as temperature or radiances that are linearly related to gray levels (see FACTOR below); enter 2 if they represent pigment concentrations (mg/m3).

(10) **FACTOR** is a parameter which is used only when the displayed image is linearly related to the gray scale, i.e. when MODE=1. If FACTOR is positive, it will represent the factor by which to divide the gray values of the image pixels in

order to convert them into actual data values. If a zero or negative number is entered, the slope and intercept for this mapping function will be obtained from the file header of each IN_FILEs. To retain the gray scale values, enter 1 (the default value); for sea surface temperature (SST), enter 8; for water radiance data, enter 85; for aerosol radiance data, enter 100.

(11) **FORMAT** specifies a FORTRAN format for the records of OU_FILEs. Generally, the default format ("0(I5)" or "1P,0(G13.5)" depending on the MODE=1 or 2) should be used. The format should have the general form of a repeat count (N) followed by a field descriptor for the numeric output types I, F, E, D, or G. A field descriptor is composed of the letter type identifier (T), a width specification (W), and, for non-integer types, a decimal specification (D). If N is not specified, it defaults to 1; if it is explicitly 0, it is set to the number of pixels processed per image line (X); if it is less than X, each image line will wrap around using as many output records as needed (see below). In addition, the scale factor (P), the plus sign control (S), and the exponent specifier (E) may also be used; if omitted, the FORTRAN defaults of 0, SS, and 2, respectively, will be used. AGAIN IT SHOULD BE EMPHASIZED THAT THE DEFAULT VALUE "0(I5)" or "1P,0(G13.5)" SHOULD BE ADEQUATE FOR MOST PURPOSES. The following table presents examples of valid FORMAT values.

FORMAT	N	T	W	D	P	S	E
------	-	-	-	-	-	-	-
1P,0(G13.5)	X	G	13	5	1	SS	2
SP,I4	1	I	4			SP	
18(F6.2)	18	F	5	2		SS	
-2P,SP,10(E16.2E4)	10	D	16	2	-2	SP	4

Parentheses and commas may be omitted, but if they are used, they must be used according to FORTRAN format syntax rules. For example, the default FORMAT, "1P,0(G13.5)", may also be written as "1P0G13.5" or "(1P0(G13.5))" but NOT as "1P0G(13.5)" or "1P0,G13.5". Slash record terminators ("/") may also be included according to FORTRAN rules. Blanks are ignored and lower or upper case letters may be used. For integer output (T="I"), the real-valued image pixels are rounded off to the nearest integer. (Please refer to the Microsoft FORTRAN manual for additional definitions and syntax rules.)

The number of output records needed for one image line will be

NR = integer((X + N - 1) / N)

where N and X are as defined above. (Note that additional records due to the use of slash record terminators in FORMAT are not included in any record calculations.) The last record of each NR set of records will contain remainder (P/N)*N pixel values. The character width of every output record will be N * W plus, if a delimiter is added, N, (e.g. for DELIM="/"),

the character width of each output record will be W*N + N). Records which do not contain N values will be filled with trailing blanks.

(12) **DELIM** may be used to insert a character (single character or a character closed by single or double quotes) for delimiter after each pixel data value field in the ASCII output files. Certain spreadsheet or statistical programs, which one may wish to use on the output files, require such delimiters. The character blank must be enclosed within quotes when entering it as value to DELIM and the character tab must be entered as ">" or ">", since the tab cannot be accepted in the input screen. If this field is blank (default), it indicates that no delimiters are to be used. Note that, in any case, blanks may be present between pixel values when the field width specified by FORMAT is large enough.

(13) **HDR_NO** is the number of header records in each of the input images (IN_FILEs). It is assumed that each input image has the same number of header records. PC-SEAPAK image files normally have one header record (the default value). These records must not be counted in any image line specification (ST_LIN, END_LN, and TOT_LN). The ASCII output files, OU_FILEs, may or may not contain any header records depending on the value of TYPE.

(14) **FLIP** defines the ASCII output file orientation. Enter one of the following values to designate the orientation of the images represented in OU_FILEs relative to those in IN_FILEs:
   0: same (the default value);
   1: reversed in the horizontal direction (mirror image);
   2: reversed in the vertical direction (upside down);
   3: reversed in the horizontal and vertical directions (mirror and upside down)

(15) **TYPE** indicates whether the ASCII output file(s), OU_FILEs, should be modified for an IBM-PC based program called SURFER. If the regular ASCII output with no header blocks is desired, one should enter a zero. On the other hand if one intends to use these files with SURFER, one should enter a value of 1. This causes the program to generate a special 5 line header required by SURFER which precedes the pixel data.

(16) **P_DIR** indicates whether one is counting pixel positions from left to right (enter 1 if so) or from right to left (in which case, enter a 2). Recall that if P_DIR = 1, the leftmost pixel is 1; if P_DIR = 2, the rightmost pixel (as defined by TOT_PX) is 1. See ST_PIX, END_PX, and TOT_PX for more information.

(17) **L_DIR** indicates whether one is counting line positions from top to bottom (enter 1 if so) or from bottom to top (in which case, enter a 2). Recall that if L_DIR = 1, the topmost line is 1; if L_DIR = 2, the bottommost line (as defined by TOT_LN) is 1. See ST_LIN, END_LN, and TOT_LN for more information.

(18) **TOT_PX** is used only if P_DIR=2. The value for this parameter will generally correspond to the width in pixels of the display system or to the rightmost pixel of the ASCII image

one is interested in producing. TOT_PX essentially enables PC-SEAPAK to determine the rightmost pixel of an image. For example, if TOT_PX=512, then the 512th pixel of each image line will be considered position 1 for the purpose of specifying ST_PIX and END_PX.

(19) **TOT_LN** is used only when L_DIR=2. The value for this parameter will generally correspond to the length in lines of the display system or to the bottommost line of the ASCII image one is interested in producing. TOT_LN essentially enables PC-SEAPAK to determine the bottommost line of an image. For example, if TOT_LN=512, then the 512th line of each image will be considered as position 1 for the purpose of specifying ST_LIN and END_LN.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: AUTOCORR
DATE: 10/28/91
MENU: STAT2

DESCRIPTION: This program plots autocorrelation or autocovariance vs. lags, of the currently displayed image over a user defined line, rectangular box, or parallelogram. The calculations can be based on gray levels, pigment concentrations, or any linearly scaled data units. A user specified range can be used to exclude land, clouds, or other invalid pixel values during the calculations. The number of lags is determined by the number of pixels along the line, along the horizontal or vertical side of the box, or along the first or second side of the parallelogram. For a box or parallelogram, the autocorrelation and the autocovariance for each lag are calculated line by line first and then averaged across all lines (i.e., all the lines are weighted equally).

PARAMETERS:

(1) **MODE** specifies the data type of the displayed image. A value of "1" (the default value) should be entered if the pixel values of the displayed image represent data (such as temperature) that are linearly related to gray levels. A value of "2" should be entered if they represent pigment concentrations (mg/m3).

(2) **FACTOR** is a non-negative scaling factor which is used only if MODE=1, i.e. the data-to-gray scale mapping function is linear for the displayed image. It is ignored when MODE=2. If FACTOR is positive, it will represent the factor by which to divide the gray values in order to convert them into actual data values. If zero is entered, the slope and intercept for this mapping function will be obtained from the header of the disk file for the displayed image. In order to retain the gray values, a "1" (the default value) should be entered ; for sea surface temperature (SST), enter 8; for water radiance data, enter 85; for aerosol radiance data, enter 100.

(3) **RANGE** defines the range of the pixel values to use for the calculations of autocorrelation or autocovariance. Two values should be entered that conform to the units of the displayed image (i.e. pigment concentration or units linearly proportional to gray levels) as specified by MODE and FACTOR. Pixel values less than the smaller RANGE value and those greater than the larger RANGE value will be excluded from the calculations. For example, to exclude land and cloud pixels for a level-2 CZCS image, the RANGE values should be 1.0 and 254.0 (the default values) for gray levels (MODE=1 and FACTOR=1) or 0.04093 and 45.0 for pigment concentrations (MODE=2).

DYNAMIC PARAMETERS:

I. Parameters for generating the plot.

(1) **OPTION** specifies whether to calculate and plot the autocorrelation ("1") or autocovariance ("2").

(2) **DIR** indicates the direction for which to calculate the autocorrelation or autocovariance and is used for a box or parallelogram only. A "1" is entered to indicate that the calculations will be along the horizontal direction of a box or in the direction of the first defined side (i.e., between the first and second corners) of a parallelogram. A "2" is entered to indicate that the calculation will be along the vertical direction of a box or the second side of a parallelogram (between the second and third corners). This parameter refers to the most recently defined area. The initial default value is 1.

(3) **G_PAL** is the graphics palette to be used for the autocorrelation or autocovariance plot.

(4) **XLABEL** is the label for the X axis of the autocorrelation or autocovariance plot and may contain up to 40 characters. Upper and lower case letters and other characters may be used. The initial default label is "LAGS"; subsequently, the previously entered label is used as the default.

(5) **YLABEL** is the label for the Y axis of the autocorrelation or autocovariance plot and may contain up to 40 characters. Upper and lower case letters and other characters may be used. The initial default label is "AUTOCORRELATION"; subsequently, the previously entered label is used as the default.

(6) **TITLE** is the title for the autocorrelation or autocovariance plot. It may contain up to 40 characters and will appear below the graph. Upper and lower case letters and other characters may be used.

II. Parameters for outputting the plot data to a file.

(1) **OPTION** specifies whether to calculate and output the autocorrelation ("1") or autocovariance ("2") plot.

(2) **O_FIL** is the output file name which will contain the lag numbers in the first column, the corresponding autocorrelation or autocovariance values in the second column, and the number of observations in the third column. A discrete character plot may also be generated, depending on the parameter PFLAG, after the third column. A name of "CON" for this parameter will send the output to the screen and "LPT1" or "LPT2" will send the output to the printer.

(3) **PFLAG** may have a value of "Y" or "N" to specify whether or not to generate a discrete character plot in the output file.

(4) **DIR** is analogous to DIR of DYNAMIC PARAMETERS I.

III. Parameters for clearing the graphics palette

(1) **CLR_PAL** is the number of the palette from which to clear overlay graphics. A "-1" is to used to clear all overlay graphics.

FUNCTION KEY DEFINITIONS:

ESC: Exits this program.

F1: Defines a line over which the autocorrelation or autocovariance values of different lags are to be obtained. More than one line segment may be used in order to approximate a curved line.

F2: Defines a box (a rectangle with horizontal and vertical sides, i.e. sides along the pixel or line direction) over which the autocorrelation or autocovariance values of different lags are to be obtained.

F3: Defines a parallelogram over which the autocorrelation or autocovariance values of different lags are to be obtained. Three corners will need to be specified in a clockwise or counter-clockwise direction using the left and/or right mouse buttons.

F4: Asks the user to enter parameters (see DYNAMIC PARAMETERS I) for autocorrelation or autocovariance calculations on different lags and plots the variogram.

F5: Outputs the plot data to the screen, the printer, or an ASCII file. The parameters of DYNAMIC PARAMETERS II will be requested.

F6: Displays the next image frame buffer.

F7: Turns all graphics palettes on/off.

F8: Turns the displayed image on/off.

F9: Increases the current graphics palette by 1 or resets it to 1 if the value is greater than 7. The current graphics palette is used for defining the line (F1), box (F2), or parallelogram (F3).

F10: Clears all the overlay graphics or a specified graphics palette. The parameter CLR_PAL will be requested.

ALT F9: Displays the current cursor position.

ALT F10: Requests new values for parameters MODE, FACTOR and RANGE (see PARAMETERS)

ALT F1: Toggles function key menu display on/off.

MOUSE RIGHT BUTTON - Toggles function key menu display on/off.

PROGRAM NAME: BATHYIMG
DATE: 10/28/91
MENU: GEOGRAPH

DESCRIPTION: This program creates a bathymetry image file from the world bathymetry data file, BATHY.DAT. This file must be stored in the directory defined by the program SPKSETUP. The file contains data at 10 minute latitude and longitude intervals. The program uses a bilinear interpolation algorithm to determine the depth of each point first, then use MIN_GRAY, MAX_GRAY, MIN_DATA and MAX_DATA to convert the depth to a gray level for the output image.

BATHY.DAT contains 1,081 ((90+90)*6+1) records, one for each 10-minute interval from -90S to 90N. Each record consists of 2,161 (360*6+1) two-byte values for 10 minute longitude intervals from 0 to 360 degrees for a total file size of 4,672,082 (2161*1081*2) bytes. A file with 5-minute resolution data is available for this program but is only distributed by request due to its large size (18,675,362 bytes).

PARAMETERS:

(1) **OUTFIL** is the name of the file to create for the bathymetry image.

(2) **NORTH** is the latitude (SOUTH to 90) to be mapped to the starting line SLIN of the output image.

(3) **SOUTH** is the latitude (-90 to NORTH) to be mapped to the ending line SLIN of the output image.

(4) **WEST** is the longitude (-180 to EAST) to be mapped to the starting pixel SPIX of the output image. Note that 0 to 360 or -180 to 180 may be used to specify longitude degrees.

(5) **EAST** is the longitude (WEST to 360) to be mapped to the ending pixel EPIX of the output image. Note that 0 to 360 or -180 to 180 may be used to specify longitude degrees.

(6) **SLIN** is the starting line (row 1 to ELIN) of the output image which corresponds to the latitude NORTH.

(7) **ELIN** is the ending line (row SLIN to 512) of the output image which corresponds to the latitude SOUTH.

(8) **SPIX** is the starting pixel (column 1 to EPIX) of the output image which corresponds to the longitude WEST.

(9) **EPIX** is the ending pixel (column SPIX to 512) of the output image which corresponds to the longitude EAST.

(10) **MIN_GRAY** is the minimum gray level value (0 to MAX_GRAY) to be assigned to the depth specified by MIN_DATA.

(11) **MAX_GRAY** is the maximum gray level value (MIN_GRAY to 255) to be assigned to the depth specified by MAX_DATA.

(12) **LAND** is the gray level value (0 to 255) to use for land areas.

(13) **MIN_DATA** is the minimum depth (in meters) to be assigned to the gray level MIN_GRAY for the output. If this value or MAX_DATA is -9999, the program will obtain the minimum value from the input file and will require a longer time to run. All data with depths between MIN_DATA and MAX_DATA will be converted to gray levels as follows:

```
SLOPE = (MAX_GRAY - MIN_GRAY) / (MAX_DATA - MIN_DATA)
GRAY  = MIN_GRAY + (DEPTH - MIN_DATA) * SLOPE
GRAY  = min(max(GRAY, MIN_GRAY), MAX_GRAY)
```

Note that all data with depths less than MIN_DATA and greater than 0 (land) will be assigned to MIN_GRAY.

(14) **MAX_DATA** is the maximum depth (in meters) to be assigned to the gray level MAX_GRAY for the output. If this value or MIN_DATA is -9999, the program will find the maximum value from the input file and will require a longer time to run. All data with depths between MIN_DATA and MAX_DATA will be converted to gray levels as follows

```
SLOPE = (MAX_GRAY - MIN_GRAY) / (MAX_DATA - MIN_DATA)
GRAY  = MIN_GRAY + (DEPTH - MIN_DATA) * SLOPE
GRAY  = min(max(GRAY, MIN_GRAY), MAX_GRAY)
```

Note that all data with depths greater than MAX_DATA will be assigned to MAX_GRAY.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: BHHIST
DATE: 10/28/91
MENU: STAT1

DESCRIPTION: This program generates the histogram data files from a set of input image files. The histogram can be collected on the full image or on blotched areas as well as on full data ranges or on specified data ranges. Each output file contains five columns of data, the histogram index, the data value, the frequency counts, the percentage of frequency, and the cumulative frequency counts. An optional histogram bar chart in text mode can also be generated in each output files.

PARAMETERS:

(1) **IMGFILS** are the input image file names to be processed. Up to 36 file names may be entered. However, since the wild card (* or ?) file format is supported, up to 300 image files can be processed. Note that if there is only one file entered for IMGFILS, the program will assume it is a text file and read the input image files from this file. Note that all the image files should have a header block.

(2) **O_EXT** is the extension name to be used for all the output file names. The output file names will be decided by replacing the path and extension in each of IMGFILS with O_PATH and O_EXT. Note that if the O_PATH is "*", then only the extension in IMGFILS will be replaced by O_EXT.

(3) **B_FIL** is the name of the blotch file that defines the image areas of interest when B_PAL is not 0. Only image data within (B_PAL greater than 0) or outside (B_PAL less than 0) of the blotch areas defined by graphics palette B_PAL will be used for histogram.

(4) **B_PAL** is the graphics palette that defines the blotches in B_FIL. The value should be in the range of -7 to 7. If the number entered is positive, pixels within the blotches will be considered. If the number is negative, pixels outside the blotches will be considered. If "0" is entered, the entire image area (512 x 512) will be used and B_FIL will be ignored.

(5) **RANGE** defines the range of IMGFILS pixel values to use for the histogram. The user should enter two values in the input data units. The RANGE values must conform to the units of the IMGFILS as specified by MODE and FACTOR (i.e. pigment concentration or units linearly proportional to gray levels). For example, to exclude land and clouds pixels, the RANGE values should be 1.0 and 254.0 for gray levels (MODE=1 and FACTOR=1) or 0.0409 and 44.46 for pigment concentrations (MODE=2).

(6) **MODE** is a flag that indicates whether the pixel values of the IMGFILS image(s) represent data (such as temperature or radiance) that are linearly related to gray levels, or pigment concentrations which are non-linear. A "1" should be entered for linear data and a "2" for pigment data.

(7) **FACTOR** is a linear scale factor used only if MODE=1, i.e. when a linear data-to-gray scale mapping function for the IMGFILS image(s) is used. If greater than zero, it will represent the factor by which to divide the gray values of IMGFILS pixels in order to convert them into actual data values; if zero is entered, the slope and intercept for this mapping function will be obtained from each file header of the IMGFILS disk image files. In order to retain the gray values, enter 1 (the default value); for sea surface temperature (SST), enter 8; for water radiance data, enter 85; for aerosol radiance data, enter 100.

(8) **CHART** is a flag to specify whether to generate the histogram bar chart in text mode on the output files. Enter 0 for No, or 1 for Yes.

(9) **O_PATH** is the path name to be used for the output files. To keep the output files as the same path of IMGFILS, the O_PATH has to be "*". Otherwise, the O_PATH specified here will replace the path specified in IMGFILS for the output files.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: BHL2MULT
DATE: 10/28/91
MENU: L2PROD

DESCRIPTION: This program is the batch run mode of the program L2MULT. It reads the input parameter sets from the parameter file one by one and generates the level-2 images for each of the input parameter sets. The parameter file must be created by the program PARL2MU. A log file which contains all the messages of the batch run can be created if desired.

PARAMETERS:
(1) **PAR_FIL** is the input parameter file name for this batch run. This file must be created by the program PARL2MU.
(2) **LOG_FIL** is the log file name. If a file name for LOG_FIL is given, then all the messages displayed on the terminal for this batch run will also be written to this file. If the LOG_FIL is blank, then no log file will be created.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: BHMAPIMG
DATE: 10/28/91
MENU: PROJECTN

DESCRIPTION: This program is the batch run mode of the program MAPIMG. It reads the input parameter sets from the parameter file one by one and generates the mapped images for each of the input parameter sets. The parameter file must be created by the program PARMPIMG. A log file which contains all the message of the batch run can be created if desired.

PARAMETERS:
(1) **PAR_FIL** is the input parameter file name for this batch run. This file must be created by the program PARMPIMG.
(2) **LOG_FIL** is the log file name. If a file name for LOG_FIL is given, then all the messages displayed on the terminal for this batch run will also be written to this file. If the LOG_FIL is blank, then no log file will be created.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: BHWTKLM
DATE: 10/28/91
MENU: L2PROD

DESCRIPTION: The program BHWTKLM is another version of the program BHL2MULT. It uses the Weitek numerical coprocessor and runs under the protected mode with the Phar Lap DOS-Extender. The program accepts the input parameter file and the log file entered by the user and then, it invokes the protected mode program WTKL2MLT.EXE (bound with DOS-Extender) using the parameter and the log file names. The program WTKL2MLT.EXE reads the input parameter sets one by one from the input parameter file and then generates the level-2 products for each of the input parameter sets. Actually, the BHWTKLM is just a driver program (run under real mode) and the main process program is the WTKL2MLT.EXE (run under protected mode).

PARAMETERS:
(1) **PAR_FIL** is the input parameter file name for this batch run. This file must be created by the program PARL2MU.
(2) **LOG_FIL** is the log file name. If a file name for LOG_FIL is given, then all the messages displayed on the terminal for this batch run will also be written to this file. If the LOG_FIL is blank, then no log file will be created.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: BHWTKMP
DATE: 10/28/91
MENU: PROJECTN

DESCRIPTION: The program BHWTKMP is another version of the program BHMAPIMG. It uses the Weitek numerical coprocessor and runs under the protected mode with the Phar Lap DOS-Extender. The program accepts the input parameter file and the log file entered by the user and then, it invokes the protected mode program WTKMPIMG.EXE (bound with DOS-Extender) using the parameter and the log file names. The program WTKMPIMG.EXE reads the input parameter sets one by one from the input parameter file and generates the mapped images for each of the input parameter sets. Actually, the BHWTKMP is just a driver program (run under real mode) and the main process program is the WTKMPIMG.EXE (run under protected mode).

PARAMETERS:

(1) **PAR_FIL** is the input parameter file name for this batch run. This file must be created by the program PARMPIMG.

(2) **LOG_FIL** is the log file name. If a file name for LOG_FIL is given, then all the messages displayed on the terminal for this batch run will also be written to this file. If the LOG_FIL is blank, then no log file will be created.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: BLOCVT
DATE: 10/28/91
MENU: VAXTOPC

DESCRIPTION: The program BLOCVT is used to convert the blotch image file from the VAX SEAPAK format into the PC-SEAPAK format. It allows the SEAPAK blotch image files created on the VAX to be used in the PC. Note that only byte values of 1, 2, 4, 8, 16, 32, 64 in the input file will be converted to byte values of 1, 2, 3, 4, 5, 6 and 7 in the output file. All other possible values between 1 and 127 created on the VAX IIS by the result of the overlapping of IIS graphics planes will be set to 0 and therefore will not be displayed on the MVP-AT. This is because the IIS on the VAX supports seven independent graphics planes which can overlap, but the MVP-AT on the PC supports seven nonindependent graphics palettes with no overlapping capability. The user should also notice that the default colors of the seven IIS graphics planes and the seven MVP-AT graphics palettes may be different. To change the default colors of the graphics palettes for the MVP-AT, the programs BPCOLOR and BP1COLOR may be used. The user can also use the program LUTMOD to do this conversion step by step. This program enables the user to convert any value (1 to 127) in the blotch created on the VAX into a value of 1 to 7 to be used in PC-SEAPAK.

PARAMETER:

(1) **VAX_BLO** is the name of the input blotch image file to be converted. This file should be created on the VAX with the SEAPAK program BPSAV and transferred to the PC.

(2) **PC_BLO** is the name of the output file containing the converted blotch to be used in PC-SEAPAK.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: BLOTCH
DATE: 10/28/91
MENU: OVERLAYS

DESCRIPTION: BLOTCH allows the user to define up to 10 polygon regions and fill them in with color. The polygons can be concave or convex, and the lines are allowed to cross. One locates the cursor at the desired starting point for the polygon and then depresses the "pick new vertex" key. After the next vertex is chosen, the "pick new vertex" key is again depressed. This defines the new vertex as well as causes a line to be drawn from the previous one. This process can be continued until the desired polygon is defined or the maximum number of vertices (500) has been reached. If necessary, the previous line may be erased by using the "delete last vertex" key. The polygon is defined after the "close region" key is depressed and it connects the first and last vertices. At this time, the user may fill the interior or exterior of the polygon region by moving the cursor inside or outside the polygon and using the "fill in blotch" key. The user may erase a filled blotch area by moving the cursor inside it and depressing the "erase blotch" key. The user also may change the graphics palette to define another blotch by using the "change graphics palette" key. There are seven graphics palettes, numbered 1 to 7, that can be used in this program with the standard default colors: red, green, yellow, blue, pink, cyan, and black. The programs GPCOLOR and BPCOLOR can be used to change these default colors. Additional function keys are provided for dropping a new image, turning the image on/off, displaying the cursor position, moving the cursor to a specified latitude/longitude position, saving the overlay blotch graphics into a file, and loading blotch graphics from a file.

PARAMETERS:
There are no parameters.

DYNAMIC PARAMETERS:

I. Used for dropping a new image (function key F7).

(1) **IMGFILE** is the name of the disk file containing the image to drop. This must be a standard 512x512x8 bit image with a known number of header blocks.

(2) **FRMBUF** is the index number (1-3) of the frame buffer to receive the image.

(3) **HEADNO** specifies the number of 512-byte header blocks in the new image. This number of blocks will be skipped before reading the image data.

(4) **YNIMG** is a flag indicating whether to display the dropped image (1) or leave on the current frame buffer being displayed (0).

II. Used for moving the cursor to a new latitude and longitude.

(1) **LAT** is the latitude where the cursor is to be moved. The value may be in decimal degrees, DMS format or radians (see parameter UNITS).

(2) **LON** is the longitude where the cursor is to be moved. The value may be in decimal degrees, DMS format or radians (see parameter UNITS)

(3) **UNITS** is the units of LAT and LON :
   1. Decimal degrees (initial default value).
   2. DMS format, sDDDMMMSSS.SS, where s is for the sign, DDD is for degrees, MMM is for minutes and SSS.SS is for seconds of an arc (for example, -75030000.00 DMS is equal to -75.5 degrees, 163006000 is equal to 163.1 degrees).
   3. Radians.

   Note that modulo arithmetic is used for all three types of units. For example, -100.0, 260.0, 620.0, etc., are all equivalent degrees and may be entered for 100 west longitude.

III. Used for saving blotch graphics into a file.

(1) **BLOFILE** is the file name in which to save all the overlay graphics created in this program. The output file will contain 512x512 bytes of data from the overlay frame buffer without any header blocks.

IV. Used for restoring blotch graphics from a file.

(1) **BLOFIL1** is the file name of the overlay graphics which are to be loaded into overlay frame buffer.

FUNCTION KEY DEFINITIONS:

ESC: Exits the program.

F1: Allows the user to select a new vertex. After the first vertex has been selected, it also causes a line to be drawn connecting the previous vertex to the present vertex.

F2: Deletes the last vertex as well as the line connecting that vertex and the previous one.

F3: Closes the polygon region by connecting the first and last vertices.

F4: Fills the interior or the exterior of a polygon depending on whether the cursor is inside or outside the polygon. The filling color is defined by current graphics palette.

F5: Erases a blotch area in which the cursor is located.

F6: Changes the overlay graphics palette by increasing the current palette number by one. If the value is greater than seven, it will be reset to one.

F7: Allows the user to drop a new image into a frame buffer.

F8: Displays the next image frame buffer.

F9: Displays the cursor's pixel and line position (TV coordinates) as well the corresponding latitude and longitude.

F10: Allows the user to move the cursor to a new position specified by a latitude and longitude.

ALT F1: Toggles the function key menu display on/off.

ALT F9: Saves the blotch graphics in the overlay frame buffer into a user specified file.

ALT F10: Loads blotch graphics from a user specified file into the overlay frame buffer.

MOUSE LEFT BUTTON - Same as function key F1.
MOUSE RIGHT BUTTON - Same as function key ALT F1.

PROGRAM NAME: BP1COLOR
DATE: 10/28/91
MENU: INITIAL

DESCRIPTION: BP1COLOR allows the user to assign a specified color to a specified output look-up table (OLUT) palette. Usually, the OLUT palette 0 is used for the LUT of the displayed image when there is an overlay. OLUT palettes 1 to 7 are used for the overlay graphics (although there are 16 palettes available for the overlay graphics, most of the time only 7 are used in PC-SEAPAK), OLUT palette 8 is used for the cursor, OLUT palettes 11 to 14 are used for the LUTs of frame buffers 0 to 3 without overlay, and OLUT palettes 17 and 18 are used for the LUTs of the overlay and displayed image when displaying the function key menu.

PARAMETERS:
(1) **PALETTE** specifies the OLUT palette (0 to 31) to be set up.
(2) **RED** specifies that all the 256 entries of the red LUT in the palette specified in parameter PALETTE will be set to this value. However, a -1 value which will set the LUT to be a linear ramp.
(3) **GREEN** specifies that all the 256 entries of the green LUT in the palette specified in parameter PALETTE will be set to this value. However, a -1 value which will set the LUT to be a linear ramp.
(4) **BLUE** specifies that all the 256 entries of the blue LUT in the palette specified in parameter PALETTE will be set to this value. However, a -1 value which will set the LUT to be a linear ramp.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: BPCOLOR
DATE: 10/28/91
MENU: INITIAL

DESCRIPTION: BPCOLOR allows the user to set up the output look-up table (OLUT) palettes 0 to 19 to specified colors. Usually, the OLUT palette 0 is used for the LUT of the displayed image when there is an overlay. OLUT palettes 1 to 7 are used for the overlay graphics (although there are 16 palettes available for the overlay graphics, most of the time only 7 are used in PC-SEAPAK), OLUT palette 8 is used for the cursor, OLUT palettes 11 to 14 are used for the LUTs of frame buffers 0 to 3 without overlay, and OLUT palettes 17 and 18 are used for the LUTs of the overlay and displayed image when displaying the function key menu. Each palette (0 to 19) has red, green and blue input entries and only 0, 1, 2 and 3 are valid inputs for those entries. The value 0 means all the 256 entries in the red, green or blue LUT will be assigned to 0 for that palette, the value 1 assigns all 256 entries to a value of 255, the value 2 assigns all 256 entries to a value of 127, the value 3 assigns the 256 entries to a linear ramp (from 0 to 255). To assign any other values (except 0, 127, 255) to all the 256 entries of red, green and blue LUTs in a palette, one can use program BP1COLOR. The default input values for the palettes in this program have been set up in the program, but the user can edit the text file PALETTE.PAR under SEAPAK directory to change the default input values.

PARAMETERS:

(1) **RED** specifies how the 256 entries of the red LUT will be set up for the palettes. A value of 0 will assign all 256 entries to 0, a value of 1 will assign the 256 entries to 255, a value of 2 will assign the 256 entries to 127, and a value of 3 will assign the 256 entries to a linear ramp.

(2) **GREEN** specifies how the 256 entries of the green LUT will be set up for the palettes. A value of 0 will assign all 256 entries to 0, a value of 1 will assign the 256 entries to 255, a value of 2 will assign the 256 entries to 127, and a value of 3 will assign the 256 entries to a linear ramp.

(3) **BLUE** specifies how the 256 entries of the blue LUT will be set up for the palettes. A value of 0 will assign all 256 entries to 0, a value of 1 will assign the 256 entries to 255, a value of 2 will assign the 256 entries to 127, and a value of 3 will assign the 256 entries to a linear ramp.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: BPLOAD
DATE: 10/28/91
MENU: OVERLAYS

DESCRIPTION: BPLOAD restores a graphics overlay onto the displayed image from a file previously created by the program BPSAV. The graphics are read into the overlay frame buffer.

PARAMETERS:
(1) **OVGRFIL** is the name of the disk file from which to input the overlay graphics.
(2) **CLEAR** is a flag indicating whether to clear any currently displayed graphics before restoring the input file's graphics ("Y") or combine the two ("N"). If "N", RFLAG will also be used.
(3) **RFLAG** is used when CLEAR="Y" to specify whether the current or restored graphics will have precedence when their pixels overlap. If "Y", graphics pixels from the file OVGRFIL will replace current graphics pixels when they occur at the same position; if "N", the current graphics pixels will remain unchanged.

FUNCTION KEY DEFINITIONS:
No function keys are used in this program.

PROGRAM NAME: BPSAV
DATE: 10/28/91
MENU: OVERLAYS

DESCRIPTION: BPSAV saves 512x512 bytes of data from the overlay frame buffer 0 into a file. For all programs which create overlay graphics on the displayed image, this program can be used to save the overlay graphics. The saved overlay graphics can then be loaded over any displayed image with the program BPLOAD.

PARAMETERS:
(1) **OVGRFILE** is the name of the disk file to create for saving the overlay graphics.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: BXCLRWAT
DATE: 10/28/91
MENU: ATMOS

DESCRIPTION: This program applies the clear water radiance algorithm to a CZCS level-1 scene in order to generate the atmospheric correction factors (in terms of "epsilons" and Angstrom exponents) used for the water radiance bands (1 to 3) in processing to level 2. The technique is to assume an initial guess for the Angstrom exponents, compute the level-2 normalized water-leaving radiance and pigment fields, eliminate pixels from consideration that do not meet specific criteria, assign the normalized water-leaving radiance at 520 nm and 550 nm of the qualifying pixels to the nominal values defined by Gordon and Clark (1981) and compute the epsilons and Angstrom exponents for those pixels.

This program uses the manual search mode (box mode) which is a modification of epsilon search techniques described in Williams et al. (1985b). There is another mode (automated mode) used by the program CLRWAT which is described elsewhere. BXCLRWAT determines "clear water" pixels inside a box by using a set of criteria (same as in CLRWAT), each of which is applied to pixels not excluded by previous criteria. The criteria and the sequence in which they are applied are as follows (ranges are inclusive):

1. Exclude land, cloud, or haze pixels: pixels whose band 5 values are greater than LANCLD (land and cloud threshold) or whose band 4 values are 255 (avoids pixels with saturated 670 nm radiances). (LANCLD is an input parameter.)
2. Exclude pixels of high sun or scanner zenith angles: pixels at which the sun zenith is greater than SUN or scanner zenith angle is greater than SCAN. (SUN and SCAN are input parameters.) This criterion is used to avoid pixels with large atmospheric path radiances which may not be accurately corrected.
3. Exclude aerosol pixels: pixels for which aerosol radiance (La(670)) values are not within the input parameter AEROL4 range. Pixels with large aerosol radiances may not be accurately corrected and pixels with low values may not contain sufficient aerosol radiance to distinguish the estimate from sensor noise.
4. Exclude pixels whose normalized band 2 or 3 water radiance (calculated using the ANGEXP input values) falls outside the NLW520 and NLW550 ranges (input parameters), respectively.
5. Exclude pixels whose pigment concentrations are greater than the input parameter PTHRES.
6. Exclude pixels for which the band 2 or 3 epsilons fall outside the EPS520 and EPS550 ranges (input parameters), respectively.
7. Exclude pixels whose epsilon values are not either monotonically increasing or decreasing. The wavelength de-

pendence on aerosol scattering should be uniformly increasing or decreasing with wavelength.

The algorithm (Gordon et al., 1983) for calculating the pigment concentrations for criterion 5 uses two-channel equations:

```
1. if Lw(550)=<0, then P=46.34456 (saturated); else,
2. if Lw(443)>0.15,
       then  P = A2 * (Lw(443)/Lw(550))**B2,          (A)
       where log10(A2)=0.053 and B2=-1.71;
       if P>=1.5 and Lw(520)>0
          then  P = A4 * (Lw(520)/Lw(550))**B4,       (B)
          where log10(A4)=0.522 and B4=-2.44;
          if P<1.5, then use (A) above;
3. if Lw(443)=<0.15 and Lw(520)>0, then use (B) above;
4. if Lw(443)=<0.15 and Lw(520)=<0,
       then P=46.34456 (saturated);
```

where Lw represents the water-leaving radiance for the band of the specified wavelength (nm) and P is the pigment concentration in mg/m^3.

The state of polarization of the light is taken into account in the calculations of the multiple Rayleigh scattering (exact radiative transfer theory; Gordon et al., 1988).

The lowest value of CLOW (epsilon(443)/La(670)) and various statistical parameters are also computed from the distribution of the Angstrom exponents of the qualifying pixels inside the box and displayed on the terminal. The statistics computed are the mean, the median, and the mean of lowest 10% of values, as well as the minimum, the maximum, the standard deviation and the quartile range (the difference between the 75th and 25th percentiles).

The user can roam the scene using a defined box area (15x15, 31x31, or 63x63 pixels) until the lowest value of CLOW has been found in a location away from the problem areas mentioned above. Since the program outputs values to the terminal, the Angstrom exponents should be noted whenever a lower CLOW is encountered.

PARAMETERS:

(1) **INFILE** is the name of an unmapped, level-1 CZCS, PC-SEAPAK image file (including the band digit). The program will need to access all five band images associated with the specified file. (Any of the five may be specified.) These files should therefore reside in the same directory.

(2) **ILTOPT** specifies the ILT option: If "1", ephemeris data from the ILT record of the level-1 scene will be used. If "0", much of these data will be obtained from the documentation record or calculated by SEAPAK based on the location and time at the start of the scene.

(3) **CORR** is the index of the correction method to use for calculating total radiances:

1: Use factors and method (time and gain dependent) of R. Evans (Univ. of Miami).

2: Use correction factors specified by FACTOR.

(4) **LANCLD** is the land/cloud threshold to identify land and cloud pixels in exclusion criterion 1. The use of this criterion in the program's algorithm is described in the main help text above.

(5) **SUN** is the solar zenith angle threshold used to avoid pixels with large atmospheric path radiances in exclusion criterion 2. The use of this criterion in the program's algorithm is described in the main help text above.

(6) **SCAN** is the scanner zenith angle threshold used to avoid pixels with large atmospheric path radiances in exclusion criterion 2. The use of this criterion in the program's algorithm is described in the main help text above.

(7) **AEROL4** is the range for valid values of aerosol radiances (La(670)) used for exclusion criterion 3. The use of this criterion in the program's algorithm is described in the main help text above.

(8) **NLW520** is the range for valid values of normalized band 2 (520nm) water radiances used for exclusion criterion 4. The use of this criterion in the program's algorithm is described in the main help text above.

(9) **NLW550** is the range for valid values of normalized band 3 (550nm) water radiances used for exclusion criterion 4. The use of this criterion in the program's algorithm is described in the main help text above.

(10) **PTHRES** is the maximum clear-water valid pigment concentration (mg/m3) used for exclusion criterion 5. The use of this criterion in the program's algorithm is described in the main help text above.

(11) **EPS520** is the range for valid values of band 2 (520nm) epsilon values used for exclusion criterion 6. The use of this criterion in the program's algorithm is described in the main help text above.

(12) **EPS550** is the range for valid values of band 3 (530nm) epsilon values used for exclusion criterion 6. The use of this criterion in the program's algorithm is described in the main help text above.

(13) **FACTOR** are the correction factors to use for calculating total radiances of bands 1 to 4, respectively. These will be used only when CORR=2.

(14) **OZONE** are the optical thicknesses (in meters) for bands 1 to 4, respectively. If the value "-999" is entered, the values used will be from the PC-TOMS database for the day of the input CZCS scene and for the point nearest to the image center. If the PC-TOMS data point is missing or an error occurs accessing the data, a message to that effect will be displayed on the terminal along with the default values. These default thicknesses are 0.00106, 0.0144, 0.0279, and 0.0125, and are the products of the absorption coefficients (3.4E-6, 46E-6, 89E-6, and 40E-6) used at the Univ. of Miami and an average amount of 313 Dobson units of ozone.

(15) **ANGEXP** are the angstrom exponents for bands 1 to 4, respectively, for use in calculating the normalized water radianc-

es and the pigment. These radiances and pigment are used in applying exclusion criteria 4 and 5. (See main help text above.)

FUNCTION KEY DEFINITIONS:

ESC: Exits the program.

F1: Calculates the Angstrom exponents on the current box.

F2: Changes the box size in a loop with sizes 15x15, 31x31 and 63x63.

F3: Changes the overlay graphics palette by increasing current palette number by one and if the value is greater than seven it will be reset to one.

F4: Marks the box at the current position in the color defined by current graphics palette.

F5: Displays the center position of the box in pixel/line coordinates and in latitude/longitude coordinates.

ALT F1: Toggles the function key menu display on/off.

MOUSE LEFT BUTTON: Same as function key F1.

MOUSE RIGHT BUTTON: Same as function key ALT F1.

PROGRAM NAME: CLR
DATE: 10/28/91
MENU: INITIAL

DESCRIPTION: The program CLR clears (erases) selected frame buffers of the MVP-AT. None of the palettes in the look-up tables are modified.

PARAMETERS:
(1) **FRMBUF** is the index (0-3) of the frame buffer to be cleared. A value of -1 may be entered to clear all four frame buffers.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: CLRWAT
DATE: 10/28/91
MENU: ATMOS

DESCRIPTION: This program applies the clear water radiance algorithm to a CZCS level-1 scene in order to generate the atmospheric correction factors (in terms of "epsilons" and Angstrom exponents) used for the water radiance bands (1 to 3) in processing to level 2. The technique is to assume an initial guess for the Angstrom exponents, compute the level-2 normalized water-leaving radiance and pigment fields, eliminate pixels from consideration that do not meet specific criteria, assign the normalized water-leaving radiance values at 520 nm and 550 nm of the qualifying pixels to the nominal values defined by Gordon and Clark (1981) and compute the epsilons and Angstrom exponents for those pixels.

This program uses the automated search mode which is a modification of epsilon search techniques described in Williams et al. (1985b). There is another mode (manual mode or box mode) used by the program BXCLRWAT which is described elsewhere. CLRWAT determines "clear water" pixels by using a set of criteria, each of which is applied to pixels not excluded by previous criteria. The criteria and the sequence in which they are applied are as follows (ranges are inclusive):

1. Exclude land, cloud, or haze pixels: pixels whose band 5 values are greater than LANCLD (land and cloud threshold) or whose band 4 values are 255 (avoids pixels with saturated 670 nm radiances). (LANCLD is an input parameter.)
2. Exclude pixels of high sun or scanner zenith angles: pixels at which the sun zenith is greater than SUN or scanner zenith angle is greater than SCAN. (SUN and SCAN are input parameters.) This criterion is used to avoid pixels with large atmospheric path radiances which may not be accurately corrected.
3. Exclude aerosol pixels: pixels for which aerosol radiance (La(670)) values are not within the input parameter AEROL4 range. Pixels with large aerosol radiances may not be accurately corrected and pixels with low values may not contain sufficient aerosol radiance to distinguish the estimate from sensor noise.
4. Exclude pixels whose normalized band 2 or 3 water radiance (calculated using the ANGEXP input values) falls outside the NLW520 and NLW550 ranges (input parameters), respectively.
5. Exclude pixels whose pigment concentrations are greater than the input parameter PTHRES.
6. Exclude pixels for which the band 2 or 3 epsilons fall outside the EPS520 and EPS550 ranges (input parameters), respectively.
7. Exclude pixels whose epsilon values are not either monotonically increasing or decreasing. The wavelength dependence

on aerosol scattering should be uniformly increasing or decreasing with wavelength.

The algorithm (Gordon et al., 1983) for calculating the pigment concentrations for criterion 5 uses two-channel equations:

```
1. if Lw(550)=<0, then P=46.34456 (saturated); else,
2. if Lw(443)>0.15,
      then  P = A2 * (Lw(443)/Lw(550))**B2,          (A)
      where log10(A2)=0.053 and B2=-1.71;
      if P>=1.5 and Lw(520)>0
         then  P = A4 * (Lw(520)/Lw(550))**B4,       (B)
         where log10(A4)=0.522 and B4=-2.44;
         if P<1.5, then use (A) above;
3. if Lw(443)=<0.15 and Lw(520)>0, then use (B) above;
4. if Lw(443)=<0.15 and Lw(520)=<0,
      then P=46.34456 (saturated);
```

where Lw represents the water-leaving radiance for the band of the specified wavelength (nm) and P is the pigment concentration in mg/m^3.

The state of polarization of the light is taken into account in the calculations of the multiple Rayleigh scattering (exact radiative transfer theory; Gordon et al., 1988).

Various statistical parameters are also computed from the distribution of the Angstrom exponents of the qualifying pixels. The statistics computed are the mean, the median, and the mean of lowest 10% of values, as well as the minimum, the maximum, the standard deviation and the quartile range (the difference between the 75th and 25th percentiles). The particular set of Angstrom exponents selected for use with the program L2MULT is up to the user. CLRWAT also creates a text file having the extension "CLR" (the root name is the filename corresponding to the input file INFILE, but without the channel number), which has all the statistical information displayed in a table. As with the other level-2 programs, a text file having the extension "L2P" and containing additional information is generated. The program also creates an image file having the extension "SCR" that contains a color code corresponding to the categories of pixels determined according to the algorithm described above. The SCR file can be displayed using the program SCREEN.

PARAMETERS:

(1) **INFILE** is the name of an unmapped, level-1 CZCS, PC-SEAPAK image file (including the band digit). The program will need to access all five band images associated with the specified file. (Any of the five may be specified.) These files should therefore reside in the same directory. Also, the INFILE name is used to formulate the names of three output files having the extensions "L2P", "CLR", and "SCR". One character prior to the INFILE extension (if any) is removed when creating these names. For example, if INFILE is "SCENE3", then SCENE.L2P, SCENE.CLR and SCENE.SCR will be created.

(2) **ILTOPT** specifies the ILT option: If "1", ephemeris data from the ILT record of the level-1 scene will be used. If "0",

much of these data will be obtained from the documentation record or calculated by SEAPAK based on the location and time at the start of the scene.

(3) **CORR** is the index of the correction method to use for calculating total radiances:
   1: Use factors and method (time and gain dependent) of R. Evans (Univ. of Miami).
   2: Use correction factors specified by FACTOR.

(4) **LANCLD** is the land/cloud threshold to identify land and cloud pixels in exclusion criterion 1. The use of this criterion in the program's algorithm is described in the main help text above.

(5) **SUN** is the solar zenith angle threshold used to avoid pixels with large atmospheric path radiances in exclusion criterion 2. The use of this criterion in the program's algorithm is described in the main help text above.

(6) **SCAN** is the scanner zenith angle threshold used to avoid pixels with large atmospheric path radiances in exclusion criterion 2. The use of this criterion in the program's algorithm is described in the main help text above.

(7) **AEROL4** is the range for valid values of aerosol radiances (La(670)) used for exclusion criterion 3. The use of this criterion in the program's algorithm is described in the main help text above.

(8) **NLW520** is the range for valid values of normalized band 2 (520nm) water radiances used for exclusion criterion 4. The use of this criterion in the program's algorithm is described in the main help text above.

(9) **NLW550** is the range for valid values of normalized band 3 (550nm) water radiances used for exclusion criterion 4. The use of this criterion in the program's algorithm is described in the main help text above.

(10) **PTHRES** is the maximum clear-water valid pigment concentration (mg/m3) used for exclusion criterion 5. The use of this criterion in the program's algorithm is described in the main help text above.

(11) **EPS520** is the range for valid values of band 2 (520nm) epsilon values used for exclusion criterion 6. The use of this criterion in the program's algorithm is described in the main help text above.

(12) **EPS550** is the range for valid values of band 3 (530nm) epsilon values used for exclusion criterion 6. The use of this criterion in the program's algorithm is described in the main help text above.

(13) **FACTOR** are the correction factors to use for calculating total radiances of bands 1 to 4, respectively. These will be used only when CORR=2.

(14) **OZONE** are the optical thicknesses (in meters) for bands 1 to 4. If the value "-999" is entered, the values used will be from the PC-TOMS database for the day of the input CZCS scene and for the point nearest to the image center. If the PC-TOMS data point is missing or an error occurs accessing the data,

a message to that effect will be displayed on the terminal along with the default values. The actual values used will be listed in the L2P log file. If defaults are used, the values will be 0.00106, 0.0144, 0.0279, and 0.0125. These thicknesses are the products of the absorption coefficients (3.4E-6, 46E-6, 89E-6, and 40E-6) used at the Univ. of Miami and an average amount of 313 Dobson units of ozone.

(15) **ANGEXP** are the angstrom exponents for bands 1 to 4, respectively, for use in calculating the normalized water radiances and the pigment. These radiances and pigment are used in applying exclusion criteria 4 and 5. (See main help text above.)

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: COAST
DATE: 10/28/91
MENU: GEOGRAPH

DESCRIPTION: This program allows the user to generate a coastline overlay for SEAPAK images using the CIA World Data Base II (WDB-II). Function key F9 is available for using an ASCII file to enter alternate or additional data to those of the CIA WDB-II. Additional function keys are available to drop new images; change the frame buffer; change, turn on/off, or erase the current graphics palette; and save the coastline graphics to a file or restore it from a file. It should be noted that the coastline is generated over the entire screen even though the image may only take up a portion of the screen.

PARAMETERS:
None.

DYNAMIC PARAMETERS:
I. Used in dropping the coastline:

(1) **INDEX** is the array of indices representing the features to be drawn over the image. The indices may be entered in any order followed by 0 to indicate that no subsequent indices are entered. Duplicate or invalid indices will be removed after input. The following indices may be specified:

- 1 major coasts/islands/lakes
- 2 additional major islands/lakes
- 3 intermediate islands/lakes
- 4 minor islands/lakes
- 6 intermittent major lakes
- 7 intermittent minor lakes
- 8 reefs
- 9 major salt pans
- 10 minor salt pans
- 13 major ice shelves
- 14 minor ice shelves
- 15 glaciers

(2) **SSRATE** is the array of the CIA WDB-II data base subsampling rate for the corresponding coastline INDEX. An SSRATE value of n indicates that only every nth data base value will be used. For most applications, the maximum value (6) should be adequate while requiring the least amount of time for the program to run.

(3) **LATRNG1** is the northern limit of the image in degrees. The default value -99999 indicates that the limit specified in the header will be used. Note that this parameter is only for the user to change the image limit specified in the header and does not set up a window for the coast line.

(4) **LATRNG2** is the southern limit of the image in degrees. The default value -99999 indicates that the limit specified in the header will be used. Note that this parameter is only for the

user to change the image limit specified in the header and does not set up a window for the coast line.

(5) **LONRNG1** is the western limit of the image in degrees. The default value -99999 indicates that the limit specified in the header will be used. Note that this parameter is only for the user to change the image limit specified in the header and does not set up a window for the coast line.

(6) **LONRNG2** is the eastern limit of the image in degrees. The default value -99999 indicates that the limit specified in the header will be used. Note that this parameter is only for the user to change the image limit specified in the header and does not set up a window for the coast line.

II. Used in dropping a new image:

(1) **IMGFILE** is the name of the disk file containing the image to drop. This must be a standard 512x512x8 bit image with a known number of header blocks.

(2) **FRMBUF** is the index number (1-3) of the frame buffer to receive the image.

(3) **HEADNO** specifies the number of 512-byte header blocks in the new image. This number of blocks will be skipped before reading the image data.

(4) **YNIMG** is a flag indicating whether to display the dropped image (1) or leave on the current frame buffer being displayed (0).

III. Used in saving the coastline graphics into a file:

(1) **BLOFILE** is the file name to save all the overlay graphics created by this program. The output file actually contains 512x512 bytes of data from the overlay frame buffer 0 and their values should always be between 0 and 7.

IV. Used in restoring the coastline graphics from a file.

(1) **BLOFIL1** is the input file name of the overlay graphics which contains 512x512 bytes of data with values between 0 and 7 and is to be loaded into the overlay frame buffer 0.

FUNCTION KEY DEFINITIONS:

ESC: Exits the program.

F1: Initiates the actual drawing of the coastline. Before the coastline is drawn, however, the user will be prompted for the coastline indices (see parameter INDEX) of the CIA WDB-II mapbase.

F2: Allows the user to drop a new image into the frame buffer.

F3: Displays the next image frame buffer.

F4: Changes the overlay graphics palette by increasing the current palette by one and is reset to one if the value is greater than seven.

F5: Toggles the current graphics palette on/off.

F6: Clears the current graphics palette.

F7: Allows the user to save the current overlay graphics data (in overlay frame buffer 0) into a file.

F8: Allows the user to restore the overlay graphics from a file into the overlay frame buffer 0.

F9: Allows the user to enter location data from an ASCII file. The data should be in the format of the following example.

```
111  33.598  130.113
111  33.581  130.098
999  9999    9999
222  33.242  134.184
...  ....    .....
```

The first, three-digit field is significant only when it contains "999," indicating the end of a series of points comprising a continuous segment to be plotted. The second and third fields are the decimal degree latitude and longitude coordinates and may be in any real format. At least one blank character must seperate the fields. The program will automatically read all data to the end of the file.

ALT F1: Toggles the function key menu display on/off.

MOUSE RIGHT BUTTON - Toggles the function key menu display on/off.

PROGRAM NAME: COLBAR
DATE: 10/28/91
MENU: LUTCOLOR

DESCRIPTION: COLBAR is a program that displays a color bar on currently displayed frame buffer. The direction (horizontal or vertical), screen location, color, size and number of blocks of the color bar can be changed by the user. A look-up table file saved using the program PAINT can be loaded into the color bar. There is a default color setup for up to 32 color blocks which is defined in the file LKTBL.PAR under the SEAPAK directory. The image with the color bar can be saved by exiting COLBAR with the color bar toggled "ON" and then saving the image using the program IMAGSAV.

PARAMETERS:

(1) **BLOCKS** is the total number of color blocks associated with the color bar to be displayed. Only values between 1 and 32 are valid input. The gray level values in the range of LGREYLEV and HGREYLEV (0 and 255 initially and can be changed using function key F8) will be evenly divided into the specified color blocks

(2) **IROW** specifies the starting line (row) on the displayed screen for the color bar.

(3) **ICOL** specifies the starting pixel (column) on the displayed screen for the color bar.

(4) **TROW** is the total number of lines (rows) for the color bar.

(5) **TCOL** is the total number of pixels (columns) for the color bar.

(6) **IDIR** defines whether the color bar will be horizontal or vertical. IDIR = 0 for horizontal and 1 for vertical.

DYNAMIC PARAMETERS:

I. Used for changing color bar location.

(1) **IROW** see PARAMETERS section.

(2) **ICOL** see PARAMETERS section.

(3) **TROW** see PARAMETERS section.

(4) **TCOL** see PARAMETERS section.

(5) **IDIR** see PARAMETERS section.

II. Used for changing gray level ranges and number of color blocks.

(1) **LGREYLEV** is the minimum gray level to be pseudocolored. The range between LGREYLEV and HGREYLEV should always be greater than or equal to the number of color blocks BLOCKS and the valid input for them are between 0 and 255 inclusive.

(2) **HGREYLEV** is the maximum gray level to be pseudocolored. The range between LGREYLEV and HGREYLEV should always be greater than or equal to the number of color blocks BLOCKS and the valid input for them are between 0 and 255 inclusive.

(3) **BLOCKS** see PARAMETERS section.

III. Used for changing the breakpoints of color blocks.

(1) **GREY_LEVEL** is an array of breakpoints to be assigned in the color blocks. The values should be entered in order. The parameters LGREYLEV and HGREYLEV will be used to check the low range of the first color block and the high range of the last color block. For example, the LGREYLEV=20, HGREYLEV=200 and the array of GREY_LEVEL entered is (10, 40, 100, 150, 220) then only four color blocks (1 to 4) will be created with gray level ranges defined as (20-40), (41-100), (101-150), (151-200).

IV. Used for saving color blocks and look-up tables.

(1) **LUTFIL2** is the output file name which the color block information and the look-up tables of red, green and blue defined by the color blocks will be saved to.

V. Used for restoring color blocks and look-up tables.

(1) **LUTFIL3** is the input look-up table file name to be loaded. Only files created using the programs PAINT or COLBAR which contain both the color block information and the look-up tables can be used.

FUNCTION KEY DEFINITIONS:

ESC: Exits the program.

F1: Allows the user to modify the colors of the color bar.

F2: Allows the user to change the color bar locations.

F3: Toggles the color bar display on/off.

F4: Toggles the image display in gray shades or in colors defined in the color bar.

F6: Allows the user to save present look-up tables which defines the currently displayed color bar into a file. The saved file can be used in programs PAINT, COLBAR and TABLOAD.

F7: Allows the user to display the color bar by retrieving the look-up tables from a file previously saved using the programs PAINT or COLBAR.

F8: Enables the user to modify the minimum and maximum gray level ranges and the number of color blocks. The valid values for the minimum and maximum gray level ranges should be between 0 and 255 and the valid value for number of color blocks should be between 1 and 32. The new specified minimum and maximum ranges will then be evenly divided into the new specified total number of color blocks.

F9: One can change the breakpoints for the color blocks using this key. After depressing this key, the user is prompted with the parameter GREY_LEVEL. The user should enter the values desired for the gray level breaks. The endpoints are assumed to be LGREYLEV and HGREYLEV and are not to be entered (key F8 can redefine this range if need be) unless blocks having single value ranges of 0 or 255 are desired. Up to 32 points may be entered. The breakpoints defined here will overwrite the setup, both the color block numbers and the breakpoints, defined in key F8.

ALT F1: Toggles the function key menu display on/off.

MOUSE RIGHT BUTTON - Same as ALT F1

PROGRAM NAME: CONTOUR
DATE: 10/28/91
MENU: OVERLAYS

DESCRIPTION: This program allows the user to contour an image at specified gray levels. The algorithm compares, horizontally and vertically, all the adjacent pixels on the image with the gray level value for contouring. If the gray level value for the contouring lies between (inclusive only on one side) the gray level values of any two adjacent pixels, the pixel with the greater gray level value will be set on the overlay frame buffer as a contour pixel. Note that this algorithm may generate isolated points as well as open contours.

The gray levels can be specified by an input value or as the value corresponding to the current cursor location. Gray level mask values, values to exclude from the contouring, can also be defined. The contours may be drawn on the full image or inside or outside blotches. Other options allow the user to fill between two contours, to change the frame buffer and the default graphics palette, to turn the overlay graphics and the displayed image on and off, and to save overlay graphics into a file or load them from a file.

PARAMETERS:
None.

DYNAMIC PARAMETERS:
I. Parameters for adding or deleting contours.
(1) **CONTOUR_VALUES** are the gray level values (0-255) for the contours to be added or deleted. Up to 10 values may be entered. Due to the algorithm (see the main help text), gray level 0 will have the same result as gray level 1.
(2) **GRAPHICS_PALETTE** are the graphics palettes (0-7) used for each corresponding contour specified in CONTOUR_VALUES. This parameter is needed only for adding contours. Note that graphics palette 0 will generate "transparent" contours, clearing any previous graphics along those lines.
II. Parameter for masking values when drawing contours.
(1) **MASK** specifies the gray level values for masking, i.e. values to ignore when drawing contours. If it is undesirable to contour in the vicinity of certain values, MASK allows the user to avoid these values. This is often helpful in obtaining clean contours when the image is noisy at the contour value. Acceptable values are 0 to 255, or -1 for none. Up to five values can be specified.
III. Parameter for clearing the graphics palette.
(1) **CLR_PAL** is the number of the palette from which to clear overlay graphics. A "-1" is to used to clear all overlay graphics.
IV. Parameter for saving overlay graphics into a file:

(1) **BLOFILE** is the file name to create for saving all overlay graphics created by this program. The output file will contain 512x512 bytes of data (values 0 to 7) from frame buffer 0.

V. Parameter for restoring overlay graphics from a file.

(1) **BLOFIL1** is the file name of the overlay graphics which contains 512x512 bytes of data (values 0 to 7) to be loaded into overlay frame buffer 0.

VI. Parameters for filling two contours

(1) **FIL_PAL** is the graphics palette (0-7) to be used for the filling.

(2) **FIL_VAL** are the two gray level values to be used in the filling process. Only the gray level values used in generating the contours (function key F1 or F2) are valid inputs. Note that the program fills all image pixels which have gray level values between (inclusive) these two values. Since the pixels on a contour may have values greater than the gray level values for that contour (see main help text), the pixels on the contours may not be filled.

VII. Parameter for full image or blotch selection

(1) **B_PAL** is the graphics palette containing the blotch areas for contouring. An integer value -7 to 7 should be entered. If the number is positive, only the pixels within the blotch will be considered; if negative, only the pixels outside the blotch will be considered. If "0" is entered, the entire image area (512 x 512) will be used.

FUNCTION KEY DEFINITIONS:

ESC: Exits the program.

F1: Allows the user to draw contours for different gray level values and on a different graphics palette. The parameters of DYNAMIC PARAMETERS I will be requested.

F2: Draws the contour with the gray level value of current cursor position and with color defined by the current graphics palette.

F3: Allows the user to delete contours created by F1 or F2. The parameter CONTOUR_VALUES of DYNAMIC PARAMETERS I will be requested.

F4: Displays current cursor position on the image and the gray level value of that pixel.

F5: Lists the gray level values and graphics palettes for the contours generated by F1 and F2.

F6: Displays the next image frame buffer.

F7: Turns all graphics palettes on/off.

F8: Turns the displayed image on/off.

F9: Increases the current graphics palette by 1 or resets it to 1 if the value is greater than 7. The current graphics palette is used for generating blotches (ALT F5) and contours with the gray level value of the current cursor position (F2).

F10: Clears all overlay graphics from a specified graphics palette. The parameter CLR_PAL will be requested.

ALT F1: Toggles the function key menu display on/off.

ALT F4: Allows the user to change the mask values when drawing contours. The parameter MASK of DYNAMIC PARAMETER II will be requested.

ALT F5: Allows the user to use the current graphics palette to define blotch areas for contouring. Once the key is pressed, a new function key set will be defined and the blotch areas created previously, if any, and the cursor will be displayed. At this time, the user can move the cursor around and press F1 or the mouse's left button to define a new vertex or F2 to erase the last vertex. As many as 500 vertices can be defined for each blotch area and up to 10 blotch areas can be defined. Key F3 is used to close the blotch area being defined and to fill the region with the color of the current graphics palette. Key F4 is used to erase a blotch area; this can only be done when the cursor is inside the blotch area. After the blotch areas are created, the ESC key must be used to return to the main function key set.

ALT F6: Redisplays the blotch areas created with ALT F5.

ALT F7: Allows the user to save the current overlay graphics data (in frame buffer 0) into a file.

ALT F8: Allows the user to restore overlay graphics from a file into frame buffer 0.

ALT F9: Allows the user to fill with color between two contours. The parameters FIL_PAL and FIL_VAL of DYNAMIC PARAMETER VI will be requested.

ALT F10: Allows the user to select the full image, or inside or outside of blotch areas for contouring. The parameter B_PAL will be requested.

MOUSE RIGHT BUTTON - Same as ALT F1.

PROGRAM NAME: CONVOLVE
DATE: 10/28/91
MENU: HARDFCT

DESCRIPTION: This program performs a transformation that gives each pixel in an image a new value that is a function of the pixels in its immediate neighborhood. The image to be transformed is in the source frame buffer SBUF and the resulting transformed image is stored in the destination frame buffer DBUF. The transformation array is user defined and stored in the parameter KERNEL. Note that if the row number KROW or column number KCOL of the kernel is even, the pixel to the top and left of center is the default central pixel. The convolution function is:

P'(x,y) = Sum of (K(i,j) * P(x+i, y+j))

where P is the SBUF pixel value at pixel (column) position x and line (row) y, K is the kernel, i ranges from -KROW/2 to KROW/2, and j ranges from -KCOL/2 to KCOL/2. The results are not normalized. The program IMGFCT may be used to perform additional arithmetic functions on the results.

PARAMETERS:

(1) **SBUF** is the frame buffer (0-3) on whose pixels the convolution will be performed.

(2) **DBUF** is the frame buffer (0-5) in which the image resulting from the convolution will be stored. (DBUF values of 4 and 5 are used to refer to frame buffers of the image memory configured as two 16-bit buffers where 4 is equivalent to the regular 8-bit buffers 0/1 and 5 is equivalent to buffers 2/3.) The results are not normalized. DBUF 4 or 5 should be used unless the user is certain that the results do not exceed 8 bits (signed). The results will be displayed on the monitor. If DBUF is 4 or 5, only the low-byte frame buffer (i.e., the regular buffers 0 or 2) of each will be displayed.

(3) **KROW** is the number of rows (2-11) to use for the kernel.

(4) **KCOL** is the number of columns (2-11) to use for the kernel.

(5) **KERNEL** is the two dimensional array of the kernel to be used for the convolution. All values must be between -127 and 127.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: CORCO
DATE: 10/28/91
MENU: STAT2

DESCRIPTION: This program calculates the correlation coefficient and other statistics for two images currently residing in two of the MVP-AT frame buffers (FBUF1 and FBUF2). The pixels used in this analysis can be the full image or the inside or outside of blotch areas and whose values are within specified ranges (RANGE1 and RANGE2). The statistical calculations include the mean values and the standard deviations for each image and the coefficients of skewness and of excess (kurtosis) for each image.

Function keys are also provided to display the scatterplot, to create or modify the blotch areas, to load a blotch from a file, to turn the displayed image or the graphics on/off, to change the current graphics palette and to change the ranges for the analyses and plots.

PARAMETERS:

(1) **FBUF1** is the frame buffer for the first input image. An integer between 1 and 3 should be entered. If a scatterplot will be requested (function key F2), the X axis of the scatterplot will be used for the image in FBUF1 and the Y axis for the FBUF2 image. Note that FBUF1 and FBUF2 cannot have the same value.

(2) **FBUF2** is the frame buffer for the second input image. An integer between 1 and 3 should be entered. If a scatterplot will be requested (function key F2), the X axis of the scatterplot will be used for the image in FBUF1 and the Y axis for the FBUF2 image. Note that FBUF1 and FBUF2 cannot have the same value.

(3) **MODE1** defines whether the image in FBUF1 is scaled linearly or is in pigment concentration. A value of "1" (the default value) should be entered if the pixel values of the FBUF1 image represent data (such as temperature) that are linearly related to gray levels. A value of "2" should be entered if they represent pigment concentrations (mg/m3).

(4) **MODE2** defines whether the image in FBUF2 is scaled linearly or is in pigment concentration. A value of "1" (the default value) should be entered if the pixel values of the FBUF2 image represent data (such as temperature) that are linearly related to gray levels. A value of "2" should be entered if they represent pigment concentrations (mg/m3).

(5) **FACTOR1** is a non-negative scaling factor which is used only if MODE1=1, i.e. the data-to-gray scale mapping function is linear for the FBUF1 image. It is ignored when MODE1=2. If FACTOR1 is positive, it will represent the factor by which to divide the gray values of FBUF1 pixels in order to convert them into actual data values. If zero is entered, the slope and intercept for this mapping function will be obtained from the header of the disk file for the FBUF1 image. In order to

retain the gray values, a "1" (the default value) should be entered; for sea surface temperature (SST), enter 8; for water radiance data, enter 85; for aerosol radiance data, enter 100. Note that the use of different linear mapping functions does not alter the appearance of the scatterplot or histograms in any way other than ensuring that the values labelling the axes reflect those of the image data.

(6) **FACTOR2** is the linear, data-to-gray scale mapping function for FBUF2. See the description of FACTOR1 whose function is analogous.

(7) **RANGE1** defines the range of FBUF1 pixel values to use for the calculation of the statistics and scatterplot. Two values should be entered for this parameter. These values should conform to the units of the FBUF1 image (i.e. pigment concentration or units linearly proportional to gray levels) as specified by MODE1 and FACTOR1. Pixel values less than the smaller RANGE1 value and those greater than the larger RANGE1 value will be excluded from the analysis and plots. (Thus pixels rejected by this criterion will also eliminate the corresponding FBUF2 pixel from consideration). For example, to exclude land and cloud pixels for a level 2 CZCS image, the RANGE1 values should be 1.0 and 254.0 (the default values) for gray levels (MODE1=1 and FACTOR1=1) or 0.04093 and 45.0 for pigment concentrations (MODE1=2).

(8) **RANGE2** defines the range for FBUF2 values in data units. See the description for RANGE1 whose function is analogous.

DYNAMIC PARAMETERS:

I. Used in statistical calculations and scatterplots.

(1) **BPAL** is the graphics palette containing the blotch area of interest. An integer in the range -7 to 7 should be entered. If the number is positive, only the pixels within the blotch will be considered; if the number is negative, only the pixels outside the blotch will be considered. If "0" is entered, the entire image area (512 x 512) will be used.

(2) **GPAL** is the graphics palette to be used to display the scatterplot. Any value from 1 to 7 may be used.

FUNCTION KEY DEFINITIONS:

ESC: Exits the program.

F1: Calculates and displays the statistical products which include the correlation coefficient between the images and the number of pixel pairs used, the mean values and standard deviations for each image, and the coefficients of skewness and of excess (kurtosis) for each image.

F2: Displays the scatterplot. The parameters BPAL and GPAL will be requested. BPAL specifies whether the full image or the inside or the outside of blotch areas will be used for the scatterplot. GPAL is the graphics palette to use for the scatterplot.

F3: Allows the user to define the blotch areas to be used for statistical calculations (F1) or scatterplot (F2). Once the

key is pressed, a new function key set will be defined and the blotch areas created previously, if any, and the cursor will be displayed. At this time, the user can move the cursor around and press F1 (or the mouse left button) to define a new vertex or F2 to erase the last vertex. As many as 500 vertices can be defined for each blotch area and up to 10 blotch areas can be defined. Key F3 is used to close the blotch area (connect the last and first vertices and fill the area with the color defined for the current graphics palette). Key F4 is used to erase a blotch area; this can only be done when the cursor is inside the blotch area. After the blotch areas are created or modified, the ESC key has to be used to return to the main function key set.

F4: Allows the user to change the input parameters MODE1, MODE2, FACTOR1, FACTOR2, RANGE1, and RANGE2.

F5: Redisplays the blotch areas defined by F3.

F6: Switches the display between the two images defined by FBUF1 and FBUF2.

F7: Turns all overlay graphics on/off.

F8: Turns the displayed image on/off.

F9: Changes the current graphics palette by increasing the palette number by one. If the value is greater than seven, it will be reset to 1.

F10: Clears the overlay graphics defined by the current graphics palette.

ALT F5: Allows the user to load the blotch areas from a file created previously using the program BLOTCH.

ALT F1: Toggles the function key menu display on/off.

MOUSE RIGHT BUTTON: Same as ALT F1.

PROGRAM NAME: DERIV
DATE: 10/28/91
MENU: SOFTFCT

DESCRIPTION: This program calculates the spatial derivatives of an input image file. One output data file (in a non-image format) is created for each derivative requested. For those output files, the program STATDIS must be used to display them.

The following derivatives may be requested:

```
DX(x,y) = (P(x,y) - P(x-1,y)) / XDIST     (1st derivative in X
                                           direction)
DY(x,y) = (P(x,y) - P(x,y-1)) / YDIST     (1st derivative in Y
                                           direction)
DXDY(x,y) = DX(x,y) + DY(x,y)             (sum of derivatives)
GRAD(x,y) = (DX(x,y)**2 + DY(x,y)**2) ** 0.5       (gradient
                                                   magnitude)
D2X(x,y) = [P(x+1,y)+P(x-1,y)-2*P(x,y)]/XDIST**2   (2nd derivative
                                                   in X direction)
D2Y(x,y) = [P(x,y+1)+P(x,y-1)-2*P(x,y)]/YDIST**2   (2nd derivative
                                                   in Y direction)
D2XY(x,y) = D2X(x,y) + D2Y(x,y)           (Laplacian)
```

where P(x,y) is the gray or pigment value at pixel x, line y.

PARAMETERS:

(1) **IFIL** is the name of the input image file to be processed.

(2) **OFIL** is the root part of the names of the output data files produced. One output file will be created for each derivative requested. The name of an output file will consist of the name specified by OFIL and a suffix corresponding to the derivative. Note that if the name of OFIL has more than four letters, only the first four letters will be used. For example, if all seven derivatives are requested and OFIL is "TEST.DAT", the following files would be created:
TESTDX.DAT,
TESTDY.DAT,
TESTDXDY.DAT,
TESTGRAD.DAT,
TESTD2X.DAT,
TESTD2Y.DAT,
and TESTD2XY.DAT.
The image files may require substantially less disk space than the corresponding data files specified by OFIL. This is illustrated by the fact that if a full image blotch is used, each of these files will require 2049 blocks (512 bytes of each block) of disk space as opposed to the normal image file of 513 blocks.

(3) **DX** is a flag indicating whether or not to calculate the first derivative in the X direction. A "1" should be entered to perform this calculation. The formula for this derivative is

DX(x,y) = (P(x,y) - P(x-1,y)) / XDIST

where P(x,y) is the gray or pigment value at pixel x, line y.

(4) **DY** is a flag indicating whether or not to calculate the first derivative in the Y direction. A "1" should be entered to perform this calculation. The formula for this derivative is

DY(x,y) = (P(x,y) - P(x,y-1)) / YDIST

where P(x,y) is the gray or pigment value at pixel x, line y.

(5) **DXDY** is a flag indicating whether or not to calculate the sum of the first partial derivatives. A "1" should be entered to perform this calculation. The formula for this derivative is

DXDY(x,y) = DX(x,y) + DY(x,y)

where P(x,y) is the gray or pigment value at pixel x, line y.

(6) **GRAD** is a flag indicating whether or not to calculate the magnitude of the gradient. A "1" should be entered to perform this calculation. The formula for this derivative is

GRAD(x,y) = (DX(x,y)**2 + DY(x,y)**2) ** 0.5

where P(x,y) is the gray or pigment value at pixel x, line y.

(7) **D2X** is a flag indicating whether or not to calculate the second derivative in the X direction. A "1" should be entered to perform this calculation. The formula for this derivative is

D2X(x,y) = [P(x+1,y)+P(x-1,y)-2*P(x,y)]/XDIST**2

where P(x,y) is the gray or pigment value at pixel x, line y.

(8) **D2Y** is a flag indicating whether or not to calculate the second derivative in the Y direction. A "1" should be entered to perform this calculation. The formula for this derivative is

D2Y(x,y) = [P(x,y+1)+P(x,y-1)-2*P(x,y)]/YDIST**2

where P(x,y) is the gray or pigment value at pixel x, line y.

(9) **D2XY** is a flag indicating whether or not to calculate the sum of the second partial derivatives (the Laplacian). A "1" should be entered to perform this calculation. The formula for this derivative.

D2XY(x,y) = D2X(x,y) + D2Y(x,y)

where P(x,y) is the gray or pigment value at pixel x, line y.

(10) **XDIST** is the distance between pixels in the X direction. The units for XDIST and YDIST must be the same.

(11) **YDIST** is the distance between pixels in the Y direction.

(12) **MODE** is a flag which indicates whether the pixel values of the IFIL image represent data (such as temperature or radiance) that are linearly related to gray levels, or pigment concentrations which are non-linear. A "1" should be entered for linear data and a "2" for pigment data.

(13) **FACTOR** is a linear scale factor used only if MODE=1, i.e. when a linear data-to-gray scale mapping function for the IFIL image is used. If greater than zero, it will represent the factor by which to divide the gray values of IFIL pixels in order to convert them into actual data values; if zero or less, the slope and intercept for this mapping function will be obtained from each file header of the IFIL disk image files. In order to retain the gray values, enter 1 (the default value); for sea surface temperature (SST), enter 8; for water radiance data, enter 85; for aerosol radiance data, enter 100.

(14) **RANGE** defines the range of IFIL pixel values to use for the calculations of derivatives. The user should enter two values in the input data units. For a given pixel location, if a value for any IFIL falls outside the RANGE values, the corresponding pixel in the output data files will be flagged as "invalid." These "invalid" pixels may be assigned any value when using STATDIS to generate the image from those output files. Again, the RANGE values must conform to the units of the IFIL image as specified by MODE and FACTOR (i.e. pigment concentration or units linearly proportional to gray levels). For example, to exclude only land and cloud pixels, the RANGE values should be 1.0 and 254.0 (the default values) for gray levels (MODE=1 and FACTOR=1) or 0.0409 and 44.46 for pigment concentrations (MODE=2).

(15) **ORIGIN** is a flag indicating the location of the origin for the image. One should enter a "0" if the origin of the image is at its top left corner or a "1" if the origin of the image is at its bottom left corner.

(16) **GPAL** is the graphics palette which defines the blotch area(s) of interest and is in the range -7 to 7. If the number entered is positive, pixels within the blotch will be considered. If the number is negative, pixels outside the blotch will be considered. Only blotches defined by this graphics palette (the absolute value of GPAL) of the blotch file BFIL will be used. If "0" is entered, the entire image area (512 x 512) will be used and BFIL will be ignored.

(17) **BFIL** is the name of the blotch file which defines the image area(s) of interest unless GPAL= 0. Only blotches defined by the graphics palette corresponding to GPAL will be used. Blotches may be drawn and saved as files using the programs BLOTCH and BPSAV.

FUNCTION KEY DEFINITIONS:

No function keys are used.

PROGRAM NAME: DIFFI
DATE: 10/28/91
MENU: SOFTFCT

DESCRIPTION: DIFFI lets you obtain a difference image file from two input image files. The differencing may be performed linearly using actual gray level units or, for images of pigment concentrations (a non-linear gray-to-data mapping), using the chlorophyll values. The differencing takes place such that input image file 2 is subtracted from input image 1. A gray level is designated by the user to represent the pixels for which the difference is zero and thus serves as an offset to the positive and negative differences. For example, for linear differencing, if a value of 128 is designated to represent this zero difference level, pixels with a difference of 5 will be displayed as gray level 133 and pixels with a difference of -5 will be displayed as gray level 123.

PARAMETERS:

(1) **INFIL1** is the file name of an image from which the image file INFIL2 is subtracted.

(2) **INFIL2** is the file name of an image to be subtracted from the file INFIL1.

(3) **OUTFIL** is the output image file name to contain the difference of INFIL1 and INFIL2.

(4) **IMODE** specifies the data mode for differencing. The user should enter a "1" to specify a linear mode or a "2" to specify a pigment mode. In linear mode, the difference is taken directly using the gray levels of each pixel of the input image files and then multiplied by WEIGHT. In pigment mode, the corresponding pigment values of each pixel of the input files are first determined. The difference is then taken in pigment values and multiplied by WEIGHT. Finally, this difference is converted back to gray levels.

(5) **ZERO** is the gray level value to be assigned to pixels with no difference between the two input images. This ZERO value serves as an offset for the positive and negative difference values. This is a very important parameter since negative difference values cannot be displayed. For example, in linear mode, if the user specifies a ZERO value of 128, all the pixels with a difference value of 50 will be displayed with the gray level 178 (128+50), while all the pixels with a difference value of -50 will be displayed with the gray level 78 (128-50). This parameter can have any integer value in the range [0,255].

(6) **WEIGHT** is the multiplicative factor for the difference, i.e. OUT=WEIGHT*(INFIL1-INFIL2)+ZERO. Note that a careful choice of the parameters WEIGHT and ZERO are required to obtain a meaningful image, i.e. one whose values stay within the screen range of [0, 255]. Values less than 0 will be set to 0 and values greater than 255 will be set to 255.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: DKAVHRR
DATE: 10/28/91
MENU: AVHRRIN

DESCRIPTION: This program is the disk version of the program TPAVHRR and ingests AVHRR LAC and GAC level 1b data files from disk instead of tape. TPAVHRR only supports the Cipher M990 tape drive with Flagstaff Engineering's tape drive software. Users with different tape drives for their PCs or on different systems must copy these files from tape to disk using the tape utilities supported by the tape drive, or download the files from another system to their PCs, before using this program.

This program supports two types of disk file formats, variable record length and fixed record length. For an AVHRR LAC tape, each file has a TBM (Terabit Memory) header record (first record, 122 bytes), a data set header record (second record, 7400 bytes), a dummy record (third record, 7400 bytes) and a variable number of data records (fourth to the end, 7400 bytes each). Each scan line is contained in two data records. The variable record length format has lengths of 122 bytes for the TBM header record and 7400 bytes for the data set header and the data records. The fixed record length format uses 7400 bytes for all records.

For an AVHRR GAC tape, each file has a TBM header record (first record, 122 bytes), a data set header record (second record, 6440 bytes) and a variable number of data records (third to the end, 6440 bytes each). Each data record contains two scan lines. The variable record length format has lengths of 122 bytes for the TBM header record and 6440 bytes for the data set header and the data records. The fixed record length format uses 6440 bytes for all records.

Depending on the tape utilities and the system, the user can select the variable length or the fixed length format to copy the tape data to disk. At the NASA/GSFC Lab. for Hydrospheric Processes, we have tested AVHRR LAC and GAC files with variable record lengths created on our PC with Cipher's M990 tape drive and Flagstaff Engineering's tape utilities, as well as with fixed record length files created on the MicroVAX II with the TU78 tape drive and MTU tape utilities and then downloaded through the network to our PC.

The format of the data in the input file must be that of tapes generated by NOAA/NESDIS/NCDC/SDSD (Kidwell, 1991). The data must be in packed format, with time incrementing, and be a full data set copy (as opposed to selective extract subsets where certain channels are selected).

PARAMETERS:

(1) **I_FIL** is the input file name. This file should be created with variable or fixed record lengths from the AVHRR LAC or GAC tape by using the tape utilities supported by the tape drive and/or the system. The program will automatically check

the file size to decide whether it has variable or fixed record lengths.

(2) **O_FIL** is the file name to use as the basis for the names of the output files created by this program. If all image files are created (see OUTPUT), and O_FIL = "AVHRR.IMG", the files created will be named as follows:

AVHRR1.IMG > Percent albedo image for visible channel 1
AVHRR2.IMG > Percent albedo image for visible channel 2
AVHRR3.IMG > Brightness energy temperature (deg C) image for IR channel 3
AVHRR4.IMG > Brightness energy temperature (deg C) image for IR channel 4
AVHRR5.IMG > Brightness energy temperature (deg C) image for IR channel 5 (if any)
AVHRR6.IMG > SST image (deg C)
AVHRR.AVH > AVHRR ingestion log file
AVHRR.CTL > Navigation control point file.

(3) **CLOUD** is the minimum percent albedo which represents clouds in the channel 1 image. Pixels whose channel 1 albedo values are greater than or equal to CLOUD will have their SST image values set to absolute white (if OUTPUT(6)="Y"). The value 100 should be used for night scenes when the channel 1 visible image is not very meaningful.

(4) **SST_EQN** specifies the index of an equation to use for calculating the sea-surface temperatures (SST) from AVHRR data. If SST_EQN=0, equation 2, 4, 5, 7, 8, 9 or 13 will be used depending upon the satellite and whether it is a day or nighttime scene. (For this purpose, the program considers ascending scenes as day scenes and descending scenes as night scenes.) Recommendations for the use of the equations with the corresponding satellite and flight direction are given below. In each equation, T(n) is the brightness energy temperature (Kidwell, 1991, p.3-14) in degrees Kelvin for AVHRR channel n and sec(SZA) is the secant of the satellite zenith angle. (References in parentheses on the right are given below.) Note that, if OUTPUT(6)="N", specifying that no SST image be created, SST_EQN will be ignored. Otherwise, pixels for which SSTs will be calculated may be restricted by the value of the CLOUD input parameter.

1. SST = T(3)*C(1) + T(4)*C(2) + T(5)*C(3) + C(4)
   where C(n) are the values of the input parameter COEFS (generalized equation).
2. SST = 1.3826*T(3) - 0.31*T(4) - 291.26
   for NOAA-6, day or night (Bernstein, 1982, p.9461).
3. SST = 1.5*(T(3)-273.15) - 0.44*(T(4)-273.15) + 1.12
   for NOAA-6, day or night (McClain, 1981, p.2).
4. SST = 1.0346*T(4) + 2.5800*(T(4)-T(5)) - 283.21
   for NOAA-7, day (Strong and McClain, 1984, p.139).
5. SST = 1.0527*T(4) + 2.6272*(T(4)-T(5)) - 288.22
   for NOAA-7, night (Barbieri et al., 1983, p.20).
6. SST = 1.0170*T(4) + 0.9700*(T(3)-T(5)) - 276.58
   for NOAA-7, night (Strong and McClain, 1984, p.139).

7. SST = 3.6569*T(4) - 2.6705*T(5) - 268.92
   for NOAA-9, day (McClain et al., 1985, p.11600).
8. SST = 3.6535*T(4) - 2.6680*T(5) - 268.41
   for NOAA-9, night (McClain et al., 1985, p.11600).
9. SST = 1.0155*T(4) + 2.5*(T(4)-T(5))
   + 0.73*(T(4)-T(5))*(sec(SZA)-1) - 277.99
   for NOAA-11, day (Kidwell, 1991).
10. SST = (T(4)-T(5)+0.789)*(0.19069*T(5)-49.16)/
   (0.20524*T(5)-0.17334*T(4)-6.78) + 0.92912*T(5)
   + 0.81*(T(4)-T(5))*(sec(SZA)-1) - 254.18
   for NOAA-11, day (Kidwell, 1991).
11. SST = (T(3)-T(5)+14.86)*(0.16835*T(4)-34.32)/
   (0.20524*T(5)-0.07747*T(3)-20.01)
   + 0.97120*T(4) + 1.87*(sec(SZA)-1) - 276.59
   for NOAA-11, night (Kidwell, 1991).
12. SST = (T(3)-T(4)-6.44)*(0.17079*T(4)-58.47)/
   (0.17334*T(4)-0.07747*T(3)-33.74)
   + 0.98530*T(4) + 1.97*(sec(SZA)-1) - 257.28
   for NOAA-11, night (Kidwell, 1991).
13. SST = (T(4)-T(5)+1.46)*(0.19596*T(5)-48.61)/
   (0.20524*T(5)-0.17334*T(4)-6.11) + 0.95476*T(5)
   + 0.98*(T(4)-T(5))*(sec(SZA)-1) - 263.84
   for NOAA-11, night (Kidwell, 1991).

(5) **COEFS** are the coefficients to use for the generalized equation for calculating sea-surface temperatures (SST). This parameter is used only if SST_EQN=1, specifying this equation which has the form

SST = T(3)*COEFS(1) + T(4)*COEFS(2) + T(5)*COEFS(3) + COEFS(4)

where T(n) is the brightness energy temperature (see Kidwell, 1991, p.3-14) in degrees Kelvin for AVHRR channel n. Note that, if OUTPUT(6)="N", specifying that no SST image be created, COEFS will be ignored even when SST_EQN=1.

(6) **SAT_NO** is the NOAA series' satellite number which is normally encoded in the file header of the scene to be ingested as described in the NOAA Polar Orbiter Data Users Guide (Kidwell, 1991). The default value of SAT_NO (0) causes the program to use this header code to determine the satellite number. If the code does not correspond to the standard NOAA code or it is otherwise incorrect, you may enter the actual satellite number for SAT_NO.

(7) **OUTPUT** allows the user to specify which SEAPAK output image to generate from the ingestion of the AVHRR level 1b tape scene. OUTPUT(n)="Y" will cause the image file corresponding to the index n to be created:

1. Percent albedo image for visible channel 1
2. Percent albedo image for visible channel 2
3. Brightness energy temperature (deg C) image for IR channel 3
4. Brightness energy temperature (deg C) image for IR channel 4

5. Brightness energy temperature (deg C) image for IR channel 5 (if any)
6. Sea-surface temperature (SST) image (deg C)

See the help text of O_FIL for more information about the convention for naming the output files.

(8) **D2GSLOPE** is the slope for converting output image data values into gray levels using the following scaling equation:

gray level = (data - D2GINTCP) * D2GSLOPE

The index of D2GSLOPE corresponds to that of OUTPUT--if OUTPUT(n)="N", D2GSLOPE(n) will be ignored. For channels 1 and 2, data are in percent albedo; for all other output images, data are in degrees centigrade.

The minimum and maximum values for all requested output images will be printed in the log file. You can use these min/max values to help you choose D2GSLOPE and D2GINTCP values that maximize the contrast of desired features in the output images when rerunning this program. However, remember that visual comparisons may be misleading for images that are of the same data type but have different scalings.

(9) **D2GINTCP** is the intercept for converting output image data values into gray levels using the following scaling equation:

gray level = (data - D2GINTCP) * D2GSLOPE

The index of D2GINTCP corresponds to that of OUTPUT--if OUTPUT(n)="N", D2GINTCP(n) will be ignored. For channels 1 and 2, data are in percent albedo; for all other output images, data are in degrees centigrade.

The minimum and maximum values for all requested output images will be printed in the log file. You can use these min/max values to help you choose D2GSLOPE and D2GINTCP values that maximize the contrast of desired features in the output images when rerunning this program. However, remember that visual comparisons may be misleading for images that are of the same data type but have different scalings.

(10) **WINDOW** defines, in conjunction with REDFAC, the area of the tape scene to use for generating the SEAPAK images. WINDOW(1) and WINDOW(3) specify the positions of the first and last samples (data points along scan lines), respectively, to use from each scan line. WINDOW(2) and WINDOW(4) specify the first and last scan line numbers to ingest from the scene.

Note that sample positions are numbered from the start of each scan. In scenes for which the satellite is ascending (flying south to north), sample 1 will be the easternmost sample of each scan; for descending scenes, it will be the westernmost sample. Also note that scan line numbers are those recorded with each scan line in the tape file of the scene. These may not start at 1 for a given scene depending on the specifications used for generating the tape file. However, scan line numbers are chronological, the lowest numbered scan line being the earliest in the scene.

To extract a certain area from an unfamiliar AVHRR scene, you may first ingest the entire scene. The data limits within the output images and whether the satellite was ascending or

descending will be indicated in the .AVH log file. These data limits are in terms of pixels and lines of an image display unit where pixel 1 is on the left and line 1 at the top. The first and last sample numbers and scan numbers and the sample and scan line reduction factors will also be found in the log file.

As an example, let's say that you have ingested this overview (entire) scene with a sample range of 1 to 2047 (LAC data) and scan line number range of 86 to 1487 with reduction factors of 4 for samples and 3 for scan lines. (Data limits would be 1 and 512 for pixels and 23 and 490 for lines.) We'll assume that, after displaying the overview, you wish to extract for greater detail a rectangular portion of data from it between pixels 100 to 150 and lines 130 to 200 (the display being 512 pixels by 512 lines).

The sample range (WINDOW(1) and WINDOW(3)) for the rectangle will be:

```
             (100)    *4 - (4-1) =  397 and
             (150)    *4 - (4-1) =  597 if descending
or       (512-150+1)*4 - (4-1) = 1449 and
         (512-100+1)*4 - (4-1) = 1649 if ascending
```

The scan line range (WINDOW(2) and WINDOW(4)) for the rectangle will be:

```
      (130- 23+1)*3 - (3-1) + (86-1) =  407 and
      (200- 23+1)*3 - (3-1) + (86-1) =  617 if descending
or    (490-200+1)*3 - (3-1) + (86-1) =  956 and
      (490-130+1)*3 - (3-1) + (86-1) = 1166 if ascending
```

The number of samples per scan line, 200, and the number of scan lines, 210, would then suggest reduction factors (REDFAC) of -2. The data in the resulting output images would be 400 pixels wide and 420 lines high.

(11) **REDFAC** are the reduction factors for the horizontal (along scan) and vertical (along orbital track) directions, in that order. Positive values indicate reduction by subsampling whereas negative values indicate magnification by pixel replication. For example, an entry of (2,2) will create images half as wide in samples and half as high in scan lines as the scene area defined by WINDOW; an entry of (-2,-2) will generate images twice as high and wide. Note that reduction in this sense indicates an increase in geographical coverage while expansion indicates a decrease. Values of -1, 0, or 1 are equivalent and generate images having a one-to-one correspondence of pixels with the tape scene.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: DKCZCS
DATE: 10/28/91
MENU: CZCSIN

DESCRIPTION: DKCZCS is the disk version of the program TPCZCS and ingests CZCS level-1 data files from disk instead of tape. TPCZCS only supports the Cipher M990 tape drive with Flagstaff Engineering's tape drive software. Users with different tape drives for their PCs or on different systems must copy these files from tape to disk using the tape utilities supported by the tape drive, or download the files from another system to their PCs, before using this program. Data copied from two types of tape formats may be used as input, the standard CZCS CRT format and the archive CZCS format.

For CRT tapes, each scene contains a header file and a data file. Since only the data file is needed, the user must skip the header file when copying the data from tape to disk. In the data file, the first and last records are documentation records of 5,328 bytes. The records in between contain up to two minutes (970 scan lines) of CZCS radiance data with record of 12,780 bytes. Tape utilities may allow the data to be copied with records of variable length or fixed length. For the variable length format, the documentation records are 5,328 bytes long and the scan line records are 12,780 bytes long. The fixed record length format has record lengths of 12,780 bytes for all records.

The archive format is created by ingesting a standard CRT tape onto disk on a VAX/VMS system. These disk files are in turn used to generate 9-track or 8mm tapes in ANSI standard or "foreign" (unlabeled) formats. (These files have records of uniform length.) The Cipher M990 tape drive with the Flagstaff Engineering tape utilities may be used for 9-track tapes; the Summus drive with its Gigasafe utilities may be used for the 8mm tapes, although these Summus utilities do not handle "foreign" tapes.

PARAMETERS:

(1) **I_FIL** is the input file name. This file should be created with variable or fixed record lengths from the CZCS level 1 tape by using the tape utilities supported by the tape drive and/or the system. The program will automatically check the file size to decide whether it has variable or fixed record length.

(2) **O_FIL** is the file name to use as the basis for the names of the output files created by this program. If O_FIL = "CZCSLV1.IMG", the files created will be named as follows:

```
CZCSLV11.IMG > For band 1 (430 nm)
CZCSLV12.IMG > For band 2 (520 nm)
CZCSLV13.IMG > For band 3 (550 nm)
CZCSLV14.IMG > For band 4 (670 nm)
CZCSLV15.IMG > For band 5 (780 nm)
CZCSLV16.IMG > For band 6 (12 microns)
CZCSLV1.CTL  > Navigation control point file.
```

(3) **WINDOW** defines, in conjunction with REDFAC, the area of the tape scene to use for generating the PC-SEAPAK images. WINDOW(1) and WINDOW(3) specify the positions of the first and last samples (data points along scan lines), respectively, to use from each scan line. Note that sample positions are numbered from the start of each scan. Since CZCS data are always collected while the satellite is ascending (flying south to north), sample 1 will be the westernmost sample of each scan. WINDOW(2) and WINDOW(4) specify the first and last scan line numbers to ingest from the scene. CZCS level-1 scenes have a maximum of 970 scan lines corresponding to two minutes of data and a fixed number of 1968 pixels per scan line. Since DKCZCS generates only 512x512 pixel image files, subsampling is required to generate an overview of the scene. Blank pixels on the side margins and lines at the bottom of an image are added if the data do not fill the 512x512 array. An error message is generated if an improper combination of reduction factors and window values is input.

(4) **REDFAC** is the reduction factors for the horizontal (along scan) and vertical (along orbital track) directions, in that order. Positive values indicate reduction by subsampling whereas negative values indicate magnification by pixel replication. For example, an entry of (2,2) will create images half as wide in samples and half as high in scan lines as the scene area defined by WINDOW; an entry of (-2,-2) will generate images twice as high and wide. Note that reduction in this sense indicates an increase in geographical coverage while expansion indicates a decrease. Values of -1, 0, or 1 are equivalent and generate images having a one-to-one correspondence of pixels with the tape scene.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: DKSDRPS
DATE: 10/28/91
MENU: AVHRRIN

DESCRIPTION: This program is the disk version of the program TPSDRPS and ingests AVHRR LAC and GAC level 1b data files in the format of those generated by NORDA/SDRPS from disk instead of from tape. TPSDRPS only supports the Cipher M990 tape drive with Flagstaff Engineering's tape drive software. Users with different tape drives for their PCs or on different systems must copy these files from tape to disk using the tape utilities supported by the tape drive, or download the files from another system to their PCs, before using this program.

For an AVHRR LAC tape, each file has a data set header record (first record, 7400 bytes), a dummy record (second record, 7400 bytes) and a variable number of data records (third to the end, 7400 bytes each). Each scan line is contained in two data records.

For an AVHRR GAC tape, each file has a data set header record (first record, 6440 bytes) and a variable number of data records (second to the end, 6440 bytes each). Each data record contains two scan lines.

For both types of tape, there may or may not be a dummy TBM file for each scene. If there is a dummy TBM file, it has to be skipped before copying the tape data to the disk. The fixed record length must be used when copying the file.

PARAMETERS:

(1) **I_FIL** is the input file name. This file should be created with fixed record lengths.

(2) **O_FIL** is the file name to use as the basis for the names of the output files created by this program. If all image files are created (see OUTPUT), and O_FIL = "AVHRR.IMG", the files created will be named as follows:

AVHRR1.IMG	>	Percent albedo image for visible channel 1
AVHRR2.IMG	>	Percent albedo image for visible channel 2
AVHRR3.IMG	>	Brightness energy temperature (deg C) image for IR channel 3
AVHRR4.IMG	>	Brightness energy temperature (deg C) image for IR channel 4
AVHRR5.IMG	>	Brightness energy temperature (deg C) image for IR channel 5 (if any)
AVHRR6.IMG	>	SST image (deg C)
AVHRR.AVH	>	AVHRR ingestion log file
AVHRR.CTL	>	Navigation control point file.

(3) **CLOUD** is the minimum percent albedo which represents clouds in the channel 1 image. Pixels whose channel 1 albedo values are greater than or equal to CLOUD will have their SST image values set to absolute white (if OUTPUT(6)="Y"). The value 100 should be used for night scenes when the channel 1 visible image is not very meaningful.

(4) **SST_EQN** specifies the index of an equation to use for calculating the sea-surface temperatures (SST) from AVHRR data. If SST_EQN=0, equation 2, 4, 5, 7, 8, 9 or 13 will be used depending upon the satellite and whether it is a day or nighttime scene. (For this purpose, the program considers ascending scenes as day scenes and descending scenes as night scenes.) Recommendations for the use of the equations with the corresponding satellite and flight direction are given below. In each equation, T(n) is the brightness energy temperature (Kidwell, 1991, p.3-14) in degrees Kelvin for AVHRR channel n and sec(SZA) is the secant of the satellite zenith angle. (References in parentheses on the right are given below.) Note that, if OUTPUT(6)="N", specifying that no SST image be created, SST_EQN will be ignored. Otherwise, pixels for which SSTs will be calculated may be restricted by the value of the CLOUD input parameter.

```
 1. SST = T(3)*C(1) + T(4)*C(2) + T(5)*C(3) + C(4)
    where C(n) are the values of the input parameter COEFS
    (generalized equation).
 2. SST = 1.3826*T(3) - 0.31*T(4) - 291.26
    for NOAA-6, day or night (Bernstein, 1982, p.9461).
 3. SST = 1.5*(T(3)-273.15) - 0.44*(T(4)-273.15) + 1.12
    for NOAA-6, day or night (McClain, 1981, p.2).
 4. SST = 1.0346*T(4) + 2.5800*(T(4)-T(5)) - 283.21
    for NOAA-7, day (Strong and McClain, 1984, p.139).
 5. SST = 1.0527*T(4) + 2.6272*(T(4)-T(5)) - 288.22
    for NOAA-7, night (Barbieri et al., 1983, p.20).
 6. SST = 1.0170*T(4) + 0.9700*(T(3)-T(5)) - 276.58
    for NOAA-7, night (Strong and McClain, 1984, p.139).
 7. SST = 3.6569*T(4) - 2.6705*T(5) - 268.92
    for NOAA-9, day (McClain et al., 1985, p.11600).
 8. SST = 3.6535*T(4) - 2.6680*T(5) - 268.41
    for NOAA-9, night (McClain et al., 1985, p.11600).
 9. SST = 1.0155*T(4) + 2.5*(T(4)-T(5))
          + 0.73*(T(4)-T(5))*(sec(SZA)-1) - 277.99
    for NOAA-11, day (Kidwell, 1991).
10. SST = (T(4)-T(5)+0.789)*(0.19069*T(5)-49.16)/
          (0.20524*T(5)-0.17334*T(4)-6.78) + 0.92912*T(5)
          + 0.81*(T(4)-T(5))*(sec(SZA)-1) - 254.18
    for NOAA-11, day (Kidwell, 1991).
11. SST = (T(3)-T(5)+14.86)*(0.16835*T(4)-34.32)/
          (0.20524*T(5)-0.07747*T(3)-20.01)
          + 0.97120*T(4) + 1.87*(sec(SZA)-1) - 276.59
    for NOAA-11, night (Kidwell, 1991).
12. SST = (T(3)-T(4)-6.44)*(0.17079*T(4)-58.47)/
          (0.17334*T(4)-0.07747*T(3)-33.74)
          + 0.98530*T(4) + 1.97*(sec(SZA)-1) - 257.28
    for NOAA-11, night (Kidwell, 1991).
13. SST = (T(4)-T(5)+1.46)*(0.19596*T(5)-48.61)/
          (0.20524*T(5)-0.17334*T(4)-6.11) + 0.95476*T(5)
          + 0.98*(T(4)-T(5))*(sec(SZA)-1) - 263.84
    for NOAA-11, night (Kidwell, 1991).
```

(5) **COEFS** are the coefficients to use for the generalized equation for calculating sea-surface temperatures (SST). This parameter is used only if SST_EQN=1, specifying this equation which has the form

SST = T(3)*COEFS(1) + T(4)*COEFS(2) + T(5)*COEFS(3) + COEFS(4)

where T(n) is the brightness energy temperature (see Kidwell, 1991, p.3-11) in degrees Kelvin for AVHRR channel n. Note that, if OUTPUT(6)="N", specifying that no SST image be created, COEFS will be ignored even when SST_EQN=1.

(6) **SAT_NO** is the NOAA series' satellite number which is normally encoded in the file header of the scene to be ingested as described in the NOAA Polar Orbiter Data Users Guide (Kidwell, 1991). The default value of SAT_NO (0) causes the program to use this header code to determine the satellite number. If the code does not correspond to the standard NOAA code or it is otherwise incorrect, you may enter the actual satellite number for SAT_NO.

(7) **OUTPUT** allows the user to specify which SEAPAK output image to generate from the ingestion of the AVHRR level 1b tape scene. OUTPUT(n)="Y" will cause the image file corresponding to the index n to be created:

1. Percent albedo image for visible channel 1
2. Percent albedo image for visible channel 2
3. Brightness energy temperature (deg C) image for IR channel 3
4. Brightness energy temperature (deg C) image for IR channel 4
5. Brightness energy temperature (deg C) image for IR channel 5 (if any)
6. Sea-surface temperature (SST) image (deg C)

See the help text of O_FIL for more information about the convention for naming the output files.

(8) **D2GSLOPE** is the slope for converting output image data values into gray levels using the following scaling equation:

gray level = (data - D2GINTCP) * D2GSLOPE

The index of D2GSLOPE corresponds to that of OUTPUT--if OUTPUT(n)="N", D2GSLOPE(n) will be ignored. For channels 1 and 2, data are in percent albedo; for all other output images, data are in degrees centigrade.

The minimum and maximum values for all requested output images will be printed in the log file. You can use these min/max values to help you choose D2GSLOPE and D2GINTCP values that maximize the contrast of desired features in the output images when rerunning this program. However, remember that visual comparisons may be misleading for images that are of the same data type but have different scalings.

(9) **D2GINTCP** is the intercept for converting output image data values into gray levels using the following scaling equation:

gray level = (data - D2GINTCP) * D2GSLOPE

The index of D2GINTCP corresponds to that of OUTPUT--if OUTPUT(n)="N", D2GINTCP(n) will be ignored. For channels 1

and 2, data are in percent albedo; for all other output images, data are in degrees centigrade.

The minimum and maximum values for all requested output images will be printed in the log file. You can use these min/max values to help you choose D2GSLOPE and D2GINTCP values that maximize the contrast of desired features in the output images when rerunning this program. However, remember that visual comparisons may be misleading for images that are of the same data type but have different scalings.

(10) **WINDOW** defines, in conjunction with REDFAC, the area of the tape scene to use for generating the SEAPAK images. WINDOW(1) and WINDOW(3) specify the positions of the first and last samples (data points along scan lines), respectively, to use from each scan line. WINDOW(2) and WINDOW(4) specify the first and last scan line numbers to ingest from the scene.

Note that sample positions are numbered from the start of each scan. In scenes for which the satellite is ascending (flying south to north), sample 1 will be the easternmost sample of each scan; for descending scenes, it will be the westernmost sample. Also note that scan line numbers are those recorded with each scan line in the tape file of the scene. These may not start at 1 for a given scene depending on the specifications used for generating the tape file. However, scan line numbers are chronological, the lowest numbered scan line being the earliest in the scene.

To extract a certain area from an unfamiliar AVHRR scene, you may first ingest the entire scene. The data limits within the output images and whether the satellite was ascending or descending will be indicated in the .AVH log file. These data limits are in terms of pixels and lines of an image display unit where pixel 1 is on the left and line 1 at the top. The first and last sample numbers and scan numbers and the sample and scan line reduction factors will also be found in the log file.

As an example, let's say that you have ingested this overview (entire) scene with a sample range of 1 to 2047 (LAC data) and scan line number range of 86 to 1487 with reduction factors of 4 for samples and 3 for scan lines. (Data limits would be 1 and 512 for pixels and 23 and 490 for lines.) We'll assume that, after displaying the overview, you wish to extract for greater detail a rectangular portion of data from it between pixels 100 to 150 and lines 130 to 200 (the display being 512 pixels by 512 lines).

The sample range (WINDOW(1) and WINDOW(3)) for the rectangle will be:

```
             (100)    *4 - (4-1) =  397 and
             (150)    *4 - (4-1) =  597 if descending
or       (512-150+1)*4 - (4-1) = 1449 and
         (512-100+1)*4 - (4-1) = 1649 if ascending
```

The scan line range (WINDOW(2) and WINDOW(4)) for the rectangle will be:

```
      (130- 23+1)*3 - (3-1) + (86-1) =  407 and
```

```
          (200- 23+1)*3 - (3-1) + (86-1) =  617 if descending
or        (490-200+1)*3 - (3-1) + (86-1) =  956 and
          (490-130+1)*3 - (3-1) + (86-1) = 1166 if ascending
```

The number of samples per scan line, 200, and the number of scan lines, 210, would then suggest reduction factors (REDFAC) of -2. The data in the resulting output images would be 400 pixels wide and 420 lines high.

(11) **REDFAC** are the reduction factors for the horizontal (along scan) and vertical (along orbital track) directions, in that order. Positive values indicate reduction by subsampling whereas negative values indicate magnification by pixel replication. For example, an entry of (2,2) will create images half as wide in samples and half as high in scan lines as the scene area defined by WINDOW; an entry of (-2,-2) will generate images twice as high and wide. Note that reduction in this sense indicates an increase in geographical coverage while expansion indicates a decrease. Values of -1, 0, or 1 are equivalent and generate images having a one-to-one correspondence of pixels with the tape scene.

FUNCTION KEY DEFINITIONS:

No function keys are used.

PROGRAM NAME: DMPHDR
DATE: 10/28/91
MENU: HEADER

DESCRIPTION: DMPHDR is used to dump all the header information stored in the first block of an image generated by PC-SEAPAK.

PARAMETER:
(1) **IMGFIL2** is the input image file name whose header is to be dumped.
(2) **OUTFIL2** is the output file name for the output listing of the header information. If this parameter is blank then all the header information will be dumped to the screen.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: DSK2DSK
DATE: 10/28/91
MENU: CZCSIN

DESCRIPTION: This program is the disk version of TP2DSK and generates full-resolution level 1 image files from disk files of CZCS level 1 data. Up to six minutes (3 scenes) of data can be ingested from three input files. Two types of output files are generated, a navigation information file and the full-resolution image files. The navigation file will have the same root name as specified in the FILENAME parameter, but with the extension ".ANC". The image files will have the name with the number 1 to N attached to the root name (not the file name extension) of FILENAME, where N is the number entered for the parameter BANDS which decides how many files will be created for bands 1 to 6. Due to the storage requirement of these image files, one should check the amount of free disk space before executing the program. A rough estimate for a single band with a full two-minute scene is 2 Mbytes. Only the program WINDOW can be used to view the output image files created by this program.

The program TP2DSK only supports the Cipher M990 tape drive with Flagstaff Engineering's tape drive software. Users with different tape drives on their PCs or on different systems must copy the CZCS level 1 data files from tape to disk using the tape utilities supported by the tape drive, or download the files from another system to their PCs, before using this program.

Data copied from two types of tape formats may be used as input, the standard CZCS CRT format and the archive CZCS format. For CRT tapes, each scene contains a header file and a data file. Since only the data file is needed, the user must skip the header file when copying the data from tape to disk. In the data file, the first and last records are documentation records of 5,328 bytes. The records in between contain up to two minutes (970 scan lines) of CZCS radiance data with record of 12,780 bytes. Tape utilities may allow the data to be copied with records of variable length or fixed length. For the variable length format, the documentation records are 5,328 bytes long and the scan line records are 12,780 bytes long. The fixed record length format has record lengths of 12,780 bytes for all records.

The archive format is created by ingesting a standard CRT tape onto disk on a VAX/VMS system. These disk files are in turn used to generate 9-track or 8mm tapes in ANSI standard or "foreign" (unlabeled) formats. (These files have records of uniform length.) The Cipher M990 tape drive with the Flagstaff Engineering tape utilities may be used for 9-track tapes; the Summus drive with its Gigasafe utilities may be used for the 8mm tapes, although these Summus utilities do not handle "foreign" tapes.

PARAMETERS:

(1) **DSKFILE** are the input disk file names. Up to three files may be entered. These files should be created from the same CZCS level 1 tape.

(2) **FULLIMG** is the name of the output file name. Numbers from 1 to N are added to the file name root (not to the extension) for each channel, where the N is the input of BANDS, for the 443, 520, 550, 670, 780 nm, and 12 micron bands, respectively.

(3) **BANDS** is the number of the output files to be created. A value of 1 to 6 should be entered. A value of 1 will only create a file for band 1 and a value of N (=2, 3, 4, 5, 6) will create N files for bands 1 to N.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: DSKLOOP
DATE: 10/28/91
MENU: FRMBUF

DESCRIPTION: This program allows the user to display up to 200 disk resident image files in a loop. The initial time interval of looping is about 1 second. The left and right mouse buttons or the up and down arrow keys may be used to increase and decrease this interval whereas the F1 function key may be used to freeze the looping at an image. Note that the minimum looping time interval cannot be reduced to less than the time to access an image file from the hard disk.

PARAMETERS:
(1) **IMGFILS** are the input disk image file names. Up to 24 names may be specified. However, since the wild cards (* or ?) may be used in the names, up to 200 image files can be looped.
(2) **HDRBLK** is the number of 512-byte header blocks preceding the data portion of each input image file.
(3) **LOOPBUF** specifies the two display frame buffers to be used during the looping. Values of 0 to 3 may be entered.

FUNCTION KEY DEFINITIONS:
ESC: Exits the program.
F1: Interrupts the looping and displays on the terminal the name of the image file displayed on the monitor. The user can press any key to resume the looping.
Up Arrow: Increases the looping time interval by about 0.2 seconds.
Down Arrow: Decreases the looping time interval by about 0.2 seconds.
Mouse Left Button: Increases the looping time interval by about 0.1 seconds.
Mouse Right Button: Decreases the looping time interval by about 0.1 second.

PROGRAM NAME: DSPIMG
DATE: 10/28/91
MENU: MIAMI

DESCRIPTION: This program generates unmapped SEAPAK image files from the bands of a level 1, 2, or 3 CZCS or AVHRR image file of the University of Miami's DSP image analysis system. The DSP image is said to be in DSP format and cannot be a binned ("PST") file. NOTE: Level 1 CZCS band images generated by DSPIMG may be used as input into SEAPAK programs (e.g. L2MULT, L2BOX) to calculate level 2 values. However, you must make sure that you run the level 2 programs without the ILT option (ILTOPT=0).

PARAMETERS:

(1) **DSP_IMG** is the input file name of the DSP image. The DSP image may be a level 1, 2, or 3 (mapped or unmapped) CZCS image or AVHRR image.

(2) **OUTROOT** is the root name to use for the disk files created. The OUTROOT with the suffix numbers 1 to N, where the N is the number of bands in the image/subimage specified by DSP_IMG and SUBIMG, will be used to form the default output file names in parameter **Output File**. For example, if OUTROOT="OUTNAME.IMG" and BANDS=("nLw440", "nLw520", "Chlor"), the output files will be: OUTNAME1.IMG, OUTNAME2.IMG, OUTNAME3.IMG and OUTNAME.CTL. NOTE: Level 1 CZCS band images generated by DSPIMG may be used as input into SEAPAK programs (e.g. L2MULT, L2BOX) to calculate level 2 values. However, you must make sure that you run the level 2 programs without the ILT option (ILTOPT=0).
The images created are equivalent to unmapped SEAPAK images of BANDS having an associated control point file regardless of whether the BANDS images are mapped or unmapped. (An unmapped SEAPAK image is one which has not been mapped using the program MAPIMG; it may or may not correspond to a satellite perspective. If the input data is pigment, they will be converted to DSP's pigment scale; otherwise, the input (DSP) calibration (scaling) will be maintained.
Unmapped DSP images may not be orientated with north toward the top and west toward the left. (For example, level 1 CZCS images may have north toward the bottom.) For such images, SEAPAK images will be created such that north is toward the top and west is toward the left but will otherwise retain the same perspective (projection). This north-at-top, west-on-left orientation is the convention for unmapped SEAPAK images and makes it is easier for users to recognize landmarks.
For DSP images that have been mapped, the SEAPAK images will have the same projection and orientation. However, even in this case the SEAPAK images will be considered to be unmapped for SEAPAK use and have an associated control point file containing the navigation information. Note that the projection will be affected if REDFAC(1) does not equal

REDFAC(2) since the aspect ratio of the output images relative to DSP_IMG will not be one.
Since only one control point file will be created for all the requested bands, if you wish to delete some of the output images, remember to retain the control point file for those undeleted images of the group.

(3) **SUB_IMG** is the subimage name in the DSP image file. Normally, "INGEST" is used as the level 1 subimage name and "CHL13" is used as the level 2 subimage name. "CHL13" may also be a level 3 subimage name. If the subimage is blank or incorrect, the first (and maybe the only) subimage will be used by default and a message will be displayed indicating the subimage name used.

(4) **REDFAC** contains the reduction factors for the pixel and line directions, in that order. Positive values indicate subsampling (reduction) whereas negative values indicate zooming-in (expansion) by pixel replication. For example, a entry of (2,2) will create images half as high and wide in pixels as (or one quarter of) the pixel area defined by WINDOW; an entry of (-2,-2) will generate images twice as high and wide. Note that reduction in this sense indicates an increase in geographical coverage while expansion indicates a decrease. Values of -1, 0, or 1 are equivalent and generate output image files having a one-to-one correspondence of pixels with DSP_IMG.

(5) **WINDOW** contains the indices of the start pixel, the start line, the end pixel, and the end line to use from DSP_IMG, in that order. For mapped DSP images, these indices refer to the location from the top and left of the DSP image. In the case of unmapped DSP images, the indices refer to the location from the top and left with the image assumed oriented with north at the top and west on the left. (Unmapped DSP images may not be orientated with north toward the top and west toward the left.)
The REDFAC values will be applied to the WINDOW indices. For example, if REDFAC=(2,2), WINDOW=(1,1,1024,1024), the top left 1024 pixel square of DSP_IMG will be used to generate 512x512 output images.

(6) **IN_RED** contains the reduction factors for the pixel and line directions, in that order, used to generate the DSP_IMG image file in Miami DSP system. These numbers are important only for unmapped, level-1 images when they are processed to level-2 using PC-SEAPAK. Note that IN_RED is not used in any way to determine the areal coverage of the output files which is controlled solely by REDFAC and WINDOW. For example, a DSP_IMG filename extension "NI7-1" implies IN_RED values of 1 (default). An extension "NI7-4" implies values of 4.

(7) **Band Names** is the list of band names in the image/subimage specified by DSPIMG and SUBIMG which may be converted into SEAPAK images. Note that the user cannot change any of the band names under this parameter.

Commonly, bands included in level 1 CZCS DSP images (e.g., files with extensions ".NI7-1", ".NI7-4") are "1", "2", "3", "4", "5", and "6", corresponding to the six CZCS channels. Level 2 bands for CZCS DSP images (e.g., extension ".FM4") include "nLw440", "nLw520", "nLw550", "La670", "K490", and "CHLOR", corresponding to the water radiance images for the first three CZCS channels and the aerosol radiance, the diffuse attenuation field, and the pigment concentration images, respectively.

(8) **Output File** contains the output file names to be created for the corresponding band names in Band Names. The default names are created by OUTROOT with the suffix numbers from the order number of the band names. The user may change the default name for each specified band name or blank it to generate no output file for that specified band.

DYNAMIC PARAMETERS:

For DSP images which are not in 8-bit form.

(1) **SCALE & BIAS** are used to convert DSP image data into SEAPAK image data. Ordinary SEAPAK images require 8-bit data to represent pixel values. If the input data are not in 8-bit form, they will be converted using the following equation:

SEAPAK_img_value = (DSP_img_value + BIAS) / 2**SCALE

The default SCALE and BIAS values will normally be 0 unless they were modified using a DSP program such as CONVRT (which may also be used to convert the DSP image to 8-bit format). For example, original AVHRR data values range from 0 to 1024, requiring up to 10 bits stored in two bytes (8 bits per byte). By using SCALE=2, the two least significant bits will be lost as all data are shifted two bits. The program will then retain the lower byte, which now contains all possible values, when generating a SEAPAK image.

The SCALE and BIAS values are incorporated into the data's slope and intercept values stored in the output file's header so that other SEAPAK programs may regenerate the original values. Of course, when SCALE>0, some loss of the original data value resolution will occur when data are regenerated in this way.

FUNCTION KEY DEFINITIONS:

No function keys are used.

PROGRAM NAME: EOF
DATE: 10/28/91
MENU: STAT2

DESCRIPTION: This program generates the EOF's (Empirical Orthogonal Function) and the time coefficient (principal component) of a sample covariance matrix. A brief matrix representation of EOF and principal component with the residual matrix is given below and the detailed algorithm is described in Murray et al. (1984).

$$V = \begin{array}{c} \textit{TIME} \\ \begin{pmatrix} v_{11} & \cdot & \cdot & \cdot & \cdot & v_{1q} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ v_{p1} & \cdot & \cdot & \cdot & \cdot & v_{pq} \end{pmatrix} \end{array} \; \textit{SPACE}$$

$$= (v_{ij}) = (x_{ij} - \bar{x}_i)$$

$$= e_1 y_1 + \ . \ . \ . \ + e_r y_r = EY$$

where

r = (rank of V) $\le \min(p,q)$

$\bar{x}_i$ = ith $(i=1,2,..,p)$ time average or row mean of (x_{ij})

x_{ij} = measurement at ith spacial point and jth time point

e_k = kth spatial EOF (column of E), orthonormal eigenvector of VV' with $e'_i e_j = \delta_{ij}$ or $E'E = I_r$ (identify matrix of order r) and V' is the transpose of V

y_k = $e'_k V$ = kth time coefficient or principal component (row of Y), orthogonal eigenvector of $V'V$ with $y_i y'_j = d_i \delta_{ij}$ or $YY' = D = \mathrm{diag}(d_1 \ldots . d_r)$

d_k = kth ordered eigenvalue of VV' (or of $V'V$) with $d_1 \ge d_2 \ge \ldots \ge d_r > 0$

The program assumes that, for most of the cases, the space dimension p is greater than the time dimension q and in order to save computation time and storage requirements it always calculates the eigenvalues (d_k's) and eigenvectors (f_k's) of V'V first. Then,

it calculates the principal components y_k by $d_k^{1/2}f'_k$ and the eigen-

vectors (e_k's) of VV' by $d_k^{-1/2}V'f_k$.

Three types of output files may be generated. The EOF files, the time averages file, and the file for eigenvalues and orthonormal eigenvectors and principal components (eigenvectors normalized to corresponding eigenvalues) of V'V. The program STATDIS has to be used to display those EOF files and the mean value file. The principal components in the third file can be plotted by using the program EOFPLOT.

PARAMETERS:

(1) **IMGFILS** are the file names of the input time series which define the time domain in the EOF processing. Up to 36 image file names may be entered. However, since the wild card (* or ?) file format is supported, up to 300 image files can be processed. Note that if there is only one file entered for IMGFILS, the program will assume it is a text file and read the input time series from this file. Note that all the image files should have a header block.

(2) **BLO_FIL** is the blotch file to be used with BPAL to define the pixels of the space domain in the EOF processing.

(3) **BPAL** is the graphics palette which defines the blotch of the area(s) of interest and is in the range -7 to 7. If the number entered is positive, pixels within the blotch will be considered. If the number is negative, pixels outside the blotch will be considered. Only blotches defined by this graphics palette (the absolute value of BPAL) in the blotch file BLO_FIL will be used. If "0" is entered, the entire image area (512x512) will be used and BLO_FIL will be ignored.

(4) **ROOT** is the root name for the output EOF files and the time averages file and the file which contains temporal eigenvalues, eigenvectors and principal components. The output EOF file names are decided by ROOT and NOUT. For example, if the name specified for ROOT is "EOF.DAT", and the NOUT specified is 5, the files "EOF1.DAT", "EOF2.DAT", "EOF3.DAT", "EOF4.DAT" and "EOF5.DAT" will be generated. These EOF files are in real data binary format and can only be displayed as an image by using the program STATDIS.
The file name "EOF.AVG" will be used for the output of the time averages. Same as the EOF files, it is in real binary data format and can be displayed as an image using the program STATDIS.
The file name "EOF.EIG" will be used for the output of the temporal eigenvalues, eigenvectors and principal components. This file is a text file and may be used in the program EOFPLOT to generate plots of those temporal components.

(5) **NOUT** is the number of output EOF files desired. This program may produce as many files as there are input images.

(6) **MODE** specifies the data type of the displayed image. A value of "1" (the default value) should be entered if the pixel values of the displayed image represent data (such as temperature) that are linearly related to gray levels. A value of "2" should be entered if they represent pigment concentrations (mg/m3).

(7) **FACTOR** is a non-negative scaling factor which is used only if MODE=1, i.e. the data-to-gray scale mapping function is linear for the displayed image. It is ignored when MODE=2. If FACTOR is positive, it will represent the factor by which to divide the gray values in order to convert them into actual data values. If zero is entered, the slope and intercept for this mapping function will be obtained from the header of the disk file for the displayed image. In order to retain the gray values, a "1" (the default value) should be entered ; for sea surface temperature (SST), enter 8; for water radiance data, enter 85; for aerosol radiance data, enter 100.

(8) **WEITEK** specifies whether the Weitek floating point coprocessor is to be used for this program. A Weitek chip has to be installed inside the computer system for WEITEK="YES". WEITEK="NO", the default, indicates that Intel 80387 coprocessor will be used to run this program.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: EOFPLOT
DATE: 10/28/91
MENU: STAT2

DESCRIPTION: This program generates the time series and power spectrum plots of a specified EOF principal component on the MVP-AT. The input file has to be created by the program EOF and has an extension of ".EIG".

The time series plot is the values of the specified principal component versus the counts of those values, i.e. the time dimension of the principal component. The power spectrum calculation is based on the maximum entropy method (MEM) from Press et al. (1986).

PARAMETERS:

I. Parameters in the first input screen.

(1) **IFIL** is the input file which contains the EOF principal component to be plotted. The file should be generated by the program EOF and have an extension of ".EIG".

(2) **ISEL** specifies the number of the principal component in the IFIL to plot. The number entered should fall in the range of 1 to the total number of the input images used in the program EOF.

(3) **OPT_T** is a flag for generating the principal component versus time series plot. The default value "1" will generate the plot, otherwise, the plot will not be generated.

(4) **COLOR_T** is the graphics palette for the principal component versus time series plot.

(5) **DFT_T** is the flag for changing the default setups for the principal component versus time series plot. The default value "1" will use the default setups for the plot, otherwise, another screen will display for user to change the positions, the lengths, the ranges, the number of tick marks and the text labels for X and Y axes of the plot.

(6) **OPT_M** is a flag for generating the power spectrum plot. The default value "1" will generate the plot, otherwise, the plot will not be generated.

(7) **COLOR_M** is the graphics palette for the power spectrum plot.

(8) **DFT_M** is the flag for changing the default setups for the power spectrum plot. The default value "1" will use the default setups for the plot, otherwise, another two screens will display for user to change coefficients for power spectrum calculation and the positions, the lengths, the number of tick marks and the text labels for X and Y axes of the plot.

II. Parameters to change the setups for the time series plot.

(1) **X-Axis** contains variables which allow the user to change the position, the length, the color, the ranges, the number of tick marks and the text label for the X axis.

(2) **Y-Axis** contains variables which allow the user to change the position, the length, the color, the ranges, the number of tick marks and the text label for the Y axis.

III. Parameters for the power spectrum calculation.

(1) **POLES** is the number of poles for the maximum entropy method (MEM) calculations. POLES may range from 1 to the number of principal component elements (N) minus 1. With larger POLES values, the spectral resolution of the power spectrum improves but spurious peaks may result and the computation time increases. As a compromise, the default value of 2*N/ln(2*N), calculated by the program, may be used. POLES represents the order of the MEM approximation equation and, as such, its value should depend on the spectral characteristics of the data. The following is from Press et al. (1986, pp. 433-434):

> In practice,...one usually wants to limit the [poles] to a few times the number of sharp spectral features... With this restricted number of poles, the method will smooth the spectrum somewhat, but this is often a desirable property... If the number of poles or the number of data points is too large, roundoff error can be a problem... With "peaky" data (i.e. data with extremely sharp spectral features), the algorithm may suggest split peaks even at modest orders, and the peaks may shift with the phase of the sine wave. Also, with noisy input functions, if you choose too high an order, you will find spurious peaks galore! Some experts recommend the use of this algorithm in conjunction with more conservative methods, like periodograms, to help choose the correct model order, and to avoid getting too fooled by spurious spectral features.

(2) **STEP** is the ratio between the time interval and the time unit of the power spectrum frequency. For example, if the time interval is one month and you would like the power spectrum frequence, to be in units of cycles/year, enter 0.0833 (=1/12).

(3) **MIN_F** is the minimum frequency (cycles per unit time) for the MEM power spectrum. MIN_F may range from 0 to 0.5/STEP (the Nyquist frequency) but must be less than MAX_F. (The actual first frequency on the spectrum plot will be the first multiple of DELTA_F greater than or equal to MIN_F.)

(4) **MAX_F** is the maximum frequency (cycles per unit time) for the MEM power spectrum. MAX_F may range from 0 to 0.5/STEP (the Nyquist frequency) but must be greater than MIN_F. (The actual last frequency on the spectrum plot will be the first multiple of DELTA_F smaller than or equal to MAX_F.)

(5) **DETREND** indicates whether or not to detrend the data linearly before MEM calculations. A "YES" or "NO" should be entered as input. The initial default value is "YES".

(6) **YSCALE** indicates the scale of the Y axis on the MEM power spectrum. A "1" should be entered for a linear scale; a "2"

should be entered for a logarithmic scale. The default value is 1.

(7) **DELTA_F** is the difference (in cycles per unit time) between adjacent frequencies for the power spectrum. The default value of DELTA_F is either (0.5/999)/DIST or (MAX_F-MIN_F)/299, whichever is greater. This default value represents the smallest value which you may use for DELTA_F. A larger value of DELTA_F saves computation time and may be adequate. Note that it is POLES, not DELTA_F, that determines the spectral resolution of the spectrum. However, DELTA_F should be small enough so that all peaks of interest are identifiable on the spectrum. After calculating the power spectrum, the total power of the spectrum and the variance of the input data will be displayed on the terminal. The total power is twice the sum of the spectral densities at the frequencies of the spectrum. These frequencies are all the multiples of DELTA_F from 0 to 0.5/DIST. Therefore, a power value much smaller than the variance indicates that significant peaks fell between spectrum frequencies and that a smaller DELTA_F should be used if possible.

IV. Parameters to change the setups for the power spectrum plot.

(1) **X-Axis** contains variables which allow the user to change the position, the length, the colors, the number of tick marks and the text label for the X axis.

(2) **Y-Axis** contains variables which allow the user to change the position, the length, the color, the number of tick marks and the text label for the Y axis.

FUNCTION KEY DEFINITIONS:

No function keys are used.

PROGRAM NAME: FILLA
DATE: 10/28/91
MENU: STAT1

DESCRIPTION: This program allows the user to replace pixel values with certain other values determined by the OPTION selection:

1: Image pixels within a blotch may be replaced by a constant.

2, 5: "Invalid" image pixels within a blotch may be replaced by the mean of surrounding "valid" pixels.

3, 6: "Invalid" image pixels within a blotch may be replaced by random values based on the mean and standard deviation of surrounding "valid" pixels.

4: Image pixels within a blotch may be replaced (filtered) by the mean of surrounding pixels.

The parameters WINDOW and VALID are used to determined the definition of "surrounding"; MIN_RNG and MAX_RNG are used to determine the "validity" of a pixel value; and BFIL and GPAL specify the "blotch" area of the image to process.

PARAMETERS:

(1) **IFIL** is the name of the image file to be processed.

(2) **OFIL** is the name of the image file to be output.

(3) **BFIL** is the name of the blotch image file defining the area(s) of interest.

(4) **OPTION** specifies any of the following options for replacing IFIL image pixel values.

1. Pixels within a blotch of a specific GPAL, and whose values are outside the corresponding MIN_RNG and MAX_RNG, will be replaced by the value of CONST.
2. Pixels within a blotch of a specific GPAL, and whose values are outside the corresponding MIN_RNG and MAX_RNG, will be replaced by the mean of the surrounding pixels. The surrounding pixels are those that are within a square window centered on the pixels to be replaced (see WINDOW). If VALID valid pixel values (as defined by the same MIN_RNG and MAX_RNG pair) are not found in this window (and VALID>0), the window size will be increased until this condition is met. Note that the mean is calculated in data units as specified by TYPE.
3. Pixels within a blotch of a specific GPAL, and whose values are outside the corresponding MIN_RNG and MAX_RNG, will be replaced by a random value based on the mean (X) and standard deviation (S) of the surrounding pixels. Those pixels are determined by WINDOW and VALID as for OPTION=2. However, X and S are calculated using the gray levels of the surrounding pixels regardless of TYPE. This is because the random values are generated from a Gaussian distribution (of X and S) and such a distribution matches more closely that of the gray values than that of pigment data. To exclude outliers as replacement values, random

values that are outside bounds are not used; the corresponding lower or upper bound on the side of the outlier is used instead. The bounds are X-S or the minimum of the surrounding pixels and X+S or the maximum of the surrounding pixels, whichever are closer to the mean. (The min/max values may be within one S of X because of the skewness of the surrounding pixels' distribution). Because outliers are rejected, the S of the actual replacement values will be somewhat smaller than that of the surrounding pixels.

4. Pixel values of pixels within the blotch area specified by any GPAL graphics palette will be replaced by the mean value of surrounding pixels. The surrounding pixels are defined by WINDOW and VALID as for OPTION=2, except that, if VALID valid values are not found, the pixel value is not replaced (i.e., the window is not expanded). Note that, as for OPTION=2, the mean is calculated in data units as specified by TYPE.
5. Same as option 2, except that if VALID valid values are not found, the window will not be expanded and the pixel value will not be replaced either.
6. Same as option 3, except that if VALID valid values are not found, the window will not be expanded and the pixel value will not be replaced either.

Note that, for OPTION 2 to 6, the MIN_RNG and MAX_RNG are used to determine which of the surrounding pixels are valid for inclusion in the calculation of a pixel replacement value.

(5) **WINDOW** specifies the width (in pixels) of a square (window) to use for determining the surrounding pixels when OPTION = 2, 3, or 4. The window will be centered over each pixel to be replaced. The MIN_RNG and MAX_RNG are then used to determine which of these pixels have "valid" values to include in the calculation of the replacement value. For OPTION = 2 or 3, if VALID such pixels are not found and VALID>0, the window width (and length) is automatically widened (lengthened) by one pixel at each end until the criterion is met. For OPTION=4, if VALID such pixels are not found, the center pixel is not changed.

(6) **VALID** is the minimum number of "valid" pixel values needed in the surrounding window for calculating replacement values for OPTION = 2, 3, or 4. (See help text for WINDOW). The MIN_RNG and MAX_RNG are used to determine the validity of the window pixels. For OPTION = 2 or 3, if VALID such pixels are not found, the window width is automatically incremented by one until the criterion is met. However, if VALID=0, the window is not enlarged (since in fact the criterion is always met) and the center pixel is not replaced if the window contains no valid pixels. For OPTION=4, if VALID such pixels are not found, the center pixel is not changed.

(7) **TYPE** indicates whether the data type is linearly related to the gray levels (TYPE=1) or pigment concentration values (TYPE=2). When TYPE=1, FACTOR is used to determine the linear data scale. TYPE is used to specify the data type the

replacement calculations of OPTION = 1, 2, and 4, and to specify the units of values entered for CONST (when OPTION=1) or MIN_RNG and MAX_RNG (when OPTION = 2, 3, or 4). It should match the type of the data in the IFIL image.

(8) **FACTOR** is used when TYPE=1 to scale the MIN_RNG and MAX_RNG values or CONST (if OPTION=1). The value entered should be the actual data to gray level mapping coefficient. For sea surface temperature data, a value of 8 should be entered. For water radiance data, a value of 85 should be entered. For aerosol radiance data, a value of 100 should be entered.

(9) **CONST** specifies the value of a constant with which to replace the pixels in the blotch area. This parameter is used only if OPTION=1. The value of CONST must be in units corresponding to the units determined by TYPE.

(10) **WEITEK** specifies whether the Weitek floating point coprocessor is to be used for this program. A Weitek chip has to be installed inside the computer system for WEITEK="YES". WEITEK="NO", the default, indicates that the Intel 80387 coprocessor will be used to run this program.

(11) **GPAL** defines the graphics palettes of BFIL which identify the areas to be processed. One can enter up to 7 integer numbers in the range of 1 to 7 for this parameter. This parameter together with the parameters MIN_RNG and MAX_RNG define the processing regions and their corresponding data processing ranges. Each graphics palette is processed independently.

(12) **MIN_RNG** and **MAX_RNG** define the range of valid pixel values for each GPAL. The values to be entered must be in the units specified by TYPE and FACTOR (if TYPE=1). For OPTION = 1, in order to specify that all pixels within a GPAL blotch are to be replaced (i.e., "invalid"), simply set the minimum value to be greater than the maximum. For OPTION = 2 or 3, pixels within a blotch of a specific GPAL, and whose values are outside the corresponding MIN_RNG and MAX_RNG pair, will be replaced. For OPTION = 2, 3, or 4, the MIN_RNG and MAX_RNG are also used to determine which of the surrounding pixels are valid for inclusion in the calculation of the replacement value for a pixel within the blotch of the corresponding GPAL.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: FILLM
DATE: 10/28/91
MENU: STAT1

DESCRIPTION: This program is used to fill land, clouds or drop-out pixels of a given image with estimated pixel values in an interactive mode under the MVP-AT. The estimated values are generated in one of two ways. The most straightforward method is to simply replace the designated pixels with a constant. The other way is via a random number method which generates a value based on the statistical distribution of a given region. For this later method, the user is requested to define two blotched regions, one for the calculation of the statistics for the random number generation and the other for the region to be filled. For the latter method, STAT_RNG is used to define the "valid" data to be included in the statistics calculations. For both of these methods, FILL_RNG is used to define the "good" data to exclude from the replacement process.

PARAMETERS:

(1) **IBUF** is the frame buffer of the image to be processed. A value of 1 to 3 should be entered.

(2) **OBUF** is the frame buffer for the output image to be generated. A value of 1 to 3 should be entered.

(3) **GPAL1** is the graphics palette to define the blotch from which the statistics are determined.

(4) **GPAL2** is the graphics palette to define the blotch on which the fill processing is to be performed.

(5) **TYPE** specifies whether the input data of STAT_RNG and FILL_RNG are linear (TYPE=1), i.e. gray level values, or pigment values (TYPE=2). Note that the values generated by the random number method based on the distribution for filling are always calculated using the gray level values regardless of TYPE. This is because the random values are generated from a Gaussian distribution (of mean X and standard deviation S) and such a distribution matches more closely that of the gray values than that of the pigment data. To exclude outliers as replacement values, random values that are outside bounds are not used; the corresponding lower or upper bound on the side of the outlier is used instead. The bounds are X-S or the minimum of the pixels in the blotch defined for mean/variance and X+S or the maximum of those pixels, whichever are closer to the mean. (The min/max values may be within one S of X because of the skewness of those pixels' distribution.) Because outliers are rejected, the S of the actual replacement values will be somewhat smaller than that of the surrounding pixels. This parameter can be changed after the program is started by pressing the function key F6.

(6) **STAT_RNG** defines the range of "valid" pixel values to be included in the statistics calculation in the blotch areas created by graphics palette GPAL1. Two values representing

the minimum and maximum valid pixel values should be entered in gray level values or pigment values depending on the TYPE specification. This parameter can be changed after the program is started by pressing the function key F6.

(7) **FILL_RNG** defines the range of "good" pixel values in the blotch areas created by graphics palette GPAL2 to exclude from the fill processing. Two values representing the minimum and maximum valid pixel values should be entered in gray level values or pigment values depending on the TYPE specification. In order to specify that all pixels within the blotch areas are to be replaced, simply set the minimum value to be greater than the maximum. This parameter can be changed after the program is started by pressing the function key F6.

DYNAMIC PARAMETERS:

I. Used in filling with constant.

(1) **TYPE** specifies whether the input data of CONST and FILL_RNG are linear (TYPE=1), i.e gray level values, or pigment values (TYPE=2).

(2) **CONST** is the value to replace image pixels within the blotch areas created with graphics palette GPAL2 and with values outside of FILL_RNG. This parameter must be in gray level or pigment value depending on the TYPE.

(3) **FILL_RNG** defines the range of "good" pixel values in the blotch areas created by graphics palette GPAL2 to exclude from the fill processing. Two values representing the minimum and maximum valid pixel values should be entered in gray level values or pigment values depending on the TYPE specification. In order to specify that all pixels within the blotch areas are to be replaced, simply set the minimum value to be greater than the maximum.

FUNCTION KEY DEFINITIONS:

ESC: Exits the program FILLM.

F1: Allows the user to use graphics palette GPAL1 to define the blotch areas for the calculation of the statistics for the random number generation. Once the key is pressed, a new function key set will be defined and the blotch areas created previously, if any, and the cursor will be displayed. At this time, the user can move the cursor around and press F1 or the mouse's left button to define a new vertex or F2 to erase the last vertex. As many as 500 vertices can be defined for each blotch area and up to 10 blotch areas can be defined. Key F3 is used to close the currently defined blotch area and fill the region with the color defined in graphics palette GPAL1. Key F4 is used to erase a blotch area; this can only be done when the cursor is inside the blotch area. After the blotch areas are created, the ESC key has to be used to return to the main function key set. The mean and the standard deviation will be calculated for IBUF pixels within this blotch and within the range specified by STAT_RNG.

F2: Allows the user to use graphics palette GPAL2 to define the blotch areas for filling in the same way as key F1 is used t define statistics area blotches on GPAL1.

F3: Asks the user to enter TYPE, CONST and FILL_RNG first and then replaces all the pixels whose values are outside of the FILL_RNG with the value CONST.

F4: Fills all the pixels in the blotch areas created by the F2 function key and have values outside of FILL_RNG with a value randomly generated based the statistics of mean and standard deviation calculated from the blotch areas created by F1 function key.

F5: Restores the image areas in OBUF which are covered by the blotch of graphics palette GPAL2 with those of the image in IBUF.

F6: Allows the user to change the TYPE, STAT_RNG and FILL_RNG.

F8: Displays the mean and standard deviation as well as the minimum and maximum values of the areas in IBUF which are covered by the blotch of graphics palette GPAL1.

F9: Toggles blotch areas defined by the graphics palette GPAL1 on/off.

F10: Toggles blotch areas defined by the graphics palette GPAL2 on/off.

ALT F1: Toggles the function key menu display on/off.

MOUSE RIGHT BUTTON: Same as ALT F1.

PROGRAM NAME: FILTER
DATE: 10/28/91
MENU: HARDFCT

DESCRIPTION: This program allows the user to perform image functions, such as filtering, edge detection, and image sharpening, that are supported by the MVP-AT.

PARAMETERS:

(1) **SBUF** is the frame buffer (0-3) on which the function SEL will be performed.

(2) **DBUF** is the frame buffer (0-3) to receive the resulting image. During the program the 16-bit frame buffer 4 (if SBUF is 0 or 1) or 5 (if SBUF is 2 or 3) will be used temporarily for the results but only the low-byte portion (i.e., the regular buffers 0 or 2) of the 16-bit frame buffer will be copied to the DBUF. (16-Bit frame buffers 4 and 5 are used to refer to buffers of the image memory configured as two 16-bit buffers where 4 is equivalent to the regular 8-bit buffers 0/1 and 5 is equivalent to buffers 2/3.)

(3) **SEL** specifies the operation to be performed on the frame buffer SBUF. Any of the following functions may be selected:

1. Low pass filtering - performs the convolution using the 3x3 kernel ((1,1,1), (1,1,1), (1,1,1)) and normalizes by dividing the resulting pixel values by 9.
2. Horizontal edge detection - performs the convolution using the 3x3 kernel ((-2,-2,-2), (0,0,0), (2,2,2)), normalizes using its absolute value, and clips to 255 so that negative and positive edges give similar results.
3. Vertical edge detection - performs the convolution using the 3x3 kernel ((-2,0,2), (-2,0,2), (-2,0,2)), normalizes using its absolute value, and clips to 255 so that negative and positive edges give similar results.
4. Prewitt edge detection - performs the convolution using two 3x3 kernels ((1,1,1), (0,0,0), (-1,-1,-1)) and ((-1,0,1), (-1,0,1), (-1,0,1)). The maximum of these values is placed in the center of the kernel.
5. Kirsch edge detection - performs horizontal, vertical and diagonal edge detection using the first four of the Kirsch compass gradient operator kernels: ((5,5,5), (-3,0,-3), (-3,-3,-3)); ((5,5,-3), (5,0,-3), (-3,-3,-3)); ((5,-3,-3), (5,0,-3), (5,-3,-3)); and ((-3,-3,-3), (5,0,-3), (5,5,-3)). After every application of each successive Kirsch operator to the image, its absolute value is computed and clipped to 255. The maximum of the obtained value and the previous maximum value is saved for comparison after the next pass.
6. Sobel edge detection - performs horizontal and vertical edges detection using the Sobel compass gradient kernels ((1,2,1), (0,0,0), (-1,-2,-1)) and ((-1,0,1), (-2,0,2),

(-1,0,1)). The maximum of this values is placed in the center of the kernel.

7. Laplacian edge detection (Type 1) - performs horizonal and vertical edge detection at the same time using the 3x3 kernel ((0,-1,0), (-1,4,-1), (0,-1,0)), normalizes using its absolute value, and clips to 255.
8. Laplacian edge detection (Type 2) - performs the convolution using the 3x3 kernel ((0,-1,0), (-1,4,-1), (0,-1,0)), normalizes using its absolute value, and clips to 255.
9. Image sharpening (Type 1) - performs the convolution using the 3x3 kernel ((0,-1,0), (-1,5,-1), (0,-1,0)), normalizes using its absolute value, and clips to 255.
10. Image sharpening (Type 2) - performs the convolution using the 3x3 kernel ((-1,-1,-1), (-1,9,-1), (-1,-1,-1)), normalizes using its absolute value, and clips to 255.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: FLAGLC
DATE: 10/28/91
MENU: CZCSL2

DESCRIPTION: This program flags land and clouds individually on an input CZCS or AVHRR image file. Two additional reference images are required. For CZCS, the corresponding CZCS level 1 channel 5 image and the corresponding CZCS level 1 channel 1 image are preferable. One of the reference images (channel 5) provides information to identify water from land and clouds while the other reference image (channel 1) provides information to identify land from clouds. The output flagged image always has cloud flagged white and land flagged to a specified gray shade (usually dark).

PARAMETERS:

(1) **LCIMG** is an image file name which can be used as a reference to identify water from land and clouds. A CZCS level 1 channel 5 image is recommended for this purpose.

(2) **CLDIMG** is an image file name which can be used as a reference to identify clouds from land. A CZCS level 1 channel 1 image is recommended for this purpose.

(3) **INIMG** is the file name of the input image to be flagged.

(4) **OUTIMG** is the image file name for the output land/cloud flagged image being generated.

(5) **SHADE** is a gray level value in the range of 0 to 255 which will be used in the output image to indicate land. Clouds are always white (the brightest gray shade 255) in the output image.

(6) **CLD_THRE** is used in conjunction with the input cloud reference image file, CLDIMG, to identify clouds from land. Any pixel in CLDIMG that has a gray value greater than the entered value is flagged as a cloud. A value in the range of 0 to 255 should be entered.

(7) **LC_THRE** is used in conjunction with the input land/cloud reference image file, LCIMG, to identify water from land/cloud. Any pixel in LCIMG that has a gray value greater than the entered value is flagged as land/clouds. A value in the range of 0 to 255 should be entered.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: GPCLR
DATE: 10/28/91
MENU: OVERLAYS

DESCRIPTION: The program GPCLR clears the overlay graphics palette. There are seven overlay graphic palettes (1-7) reserved for PC-SEAPAK. To display an image with a graphics overlay, the palette(s) (1-7) to be used should be cleared. Note, if the current image display has no overlay, then the program has no effect.

PARAMETERS:
(1) **GC_PAL** is the palette number to be cleared in the overlay frame buffer 0. The palettes 1 to 7 are reserved for overlay graphics and the palette 8 is reserved for cursor. Only integer numbers 1 to 8 are valid inputs, a value -1 will clear all graphics palettes from 1 to 8.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: GPCOLOR
DATE: 10/28/91
MENU: INITIAL

DESCRIPTION: GPCOLOR allows the user to change the colors assigned to the graphics palettes. There are 17 color codes from which to choose.

PARAMETERS:

(1) **PAL_NO** are the graphics palettes to change. Palette numbers 1 to 7 may be specified in any order. An entry of -1 signals the end of the list.

(2) **COLOR** is a list of colors to assign to the graphics palettes listed in PAL_NO. Any of the following two-letter color codes may be entered:

RE	red	BR	brown
GR	green	PI	pink
BL	blue	GY	gray
YE	yellow	WH	white
CY	cyan	MA	magenta
OR	orange	TA	tan
SA	sand	LG	light green
BK	black	OF	transparent (no color)
CL	erase graphics		

Note that if an invalid color is entered, no action will be taken.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: GPONOFF
DATE: 10/28/91
MENU: OVERLAYS

DESCRIPTION: This is a simple program to turn any graphics overlays on or off.

PARAMETERS:
(1) **OFLAG** is a flag to indicate whether to turn the overlay graphics on (1) or off (0).
(2) **OVBUF** is the overlay frame buffer. In PC-SEAPAK, this would normally be 0.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: GRID
DATE: 10/28/91
MENU: GEOGRAPH

DESCRIPTION: This program will generate a geocoordinate grid with annotation over a displayed image. The image must be a valid PC-SEAPAK image. There are no function keys associated with this program. See Appendix for a description fonts.

PARAMETERS:

MAIN MENU PARAMETERS

(1) **GRID** determines if a grid is to be overlaid on the current image. A "Y" indicates that a grid is desired. A "Y" value for LATS or LONS is not required in order to generate a grid.

(2) **DFT_G** determines if default values for the grid are to be used. A "Y" instructs the program to use default values for all input parameters associated with the grid lines. A prompt for these parameters will not be issued. DFT_G is ignored if GRID = "N".

(3) **LONS** determines if longitude labels are to be generated. A "Y" indicates that longitude labels are desired. A "Y" value for GRID is not required in order to generate the labels.

(4) **LATS** determines if latitude labels are to be generated. A "Y" indicates that latitude labels are desired. A "Y" value for GRID is not required in order to generate the labels.

(5) **DFT_L** determines if default values for the labels are to be used. A "Y" instructs the program to use default values for all input parameters associated with the grid labels. Prompts for these parameters will not be issued. DFT_L is ignored if LATS and LONS are "N".

DYNAMIC GRID PARAMETERS

(6) **LON_RNGE** is the longitudinal limits of the grid in degrees. The western limit must be the first value. If the default value -9999.0 is entered, the limits specified in the header of the image will be used. The western limit is used along with the specified spacing (see LON_DELT) and WINDOW to determine which longitudes are displayed. Therefore, if the western limit longitude falls within WINDOW, it will be displayed. However, if the eastern limit longitude falls within WINDOW, it may or may not be displayed depending on the spacing.

(7) **LON_DELT** is the spacing (in degrees) are to be displayed. Longitudes to be displayed will be the western limit (see LON_RNGE) and all subsequent longitudes which are LON_DELT degrees apart up to the eastern limit. Therefore, the eastern limit longitude will be displayed only if the longitudinal distance between the limits is a multiple of LON_DELT. Longitudes actually displayed must also be within the area defined by WINDOW. If the value -9999.0 (default) is entered,

a reasonable value will be used based on the longitudinal limits.

(8) **LON_INCR** is used in the calculating the position of various longitude points along that latitude. LON_INCR (LONgitude_-INCRement) is used to specify the distance in degrees between those longitude points and thus controls how smooth the latitudes appear when drawn. If the value -9999.0 (default) or a value larger than that specified by LON_DELT is entered, the LON_DELT spacing will be used for LON_INCR. Obviously, if GTYPE=2, LON_INCR is not used since only the latitude/ longitude intersections are marked.

(9) **LAT_RNGE** is the latitudinal limits of the grid in degrees. If the value -9999.0 (default) is entered, the limits specified in the image's header will be used. The southern limit is used along with the specified spacing (see LAT_DELT) and WINDOW to determine which latitudes are displayed. Therefore, if the southern limit latitude falls within WINDOW, it will be displayed. However, if the northern limit latitude falls within WINDOW, it may or may not be displayed depending on the spacing.

(10) **LAT_DELT** is the spacing (in degrees) between latitudes which are to be displayed. Latitudes to be displayed will be the southern limit (see LON_RNGE) and all subsequent latitudes which are LAT_DELT degrees apart up to the northern limit. Therefore, the northern limit latitude will be displayed only if the latitudinal distance between the limits is a multiple of LAT_DELT. Latitudes actually displayed must also be within the area defined by WINDOW. If the default value -9999.0 is entered, a reasonable value will be used based on the latitudinal limits.

(11) **LAT_INCR** is used in calculating the position of various latitude points along that longitude. LAT_INCR (LATitude_-INCRement) is used to specify the distance in degrees between those latitude points and thus controls how smooth the longitudes appear when drawn. If a value -9999.0 (default) or a value larger than that specified by LAT_DELT is entered, the LAT_DELT spacing will be used for LAT_INCR. Obviously, if GTYPE=2, LAT_INCR is not used since only the latitude/ longitude intersections are marked.

(12) **G_PAL** is the value of the graphics palette to use for the grid and border. If the value is negative, any graphics created with that graphics palette will first be deleted. Graphics palette 1 (positive) is the default value.

(13) **GTYPE** determines the type of line to be drawn. If GTYPE=1 (default), the latitudes/longitudes to be displayed will be drawn as lines. If GTYPE=2, only the intersections of these latitudes/longitudes will be marked by plus signs ("+").

(14) **WINDOW** defines a rectangular view area which may be all or part of the entire image display area. Only grid lines falling within this view area will be displayed. The WINDOW values represent the start pixel, end pixel, start line, and end line numbers of the view area in that order. The maximum

display area for the MVP-AT image processor board is 512 pixels wide by 512 lines high and is used by default.

(15) **BORDER** determines if a line is to be drawn around the view area. If "YES", straight lines will be drawn around the view area defined by the WINDOW grid parameter. "NO" is the default value.

DYNAMIC LABEL PARAMETERS

Longitude Label Parameters:

(16) **LON_RG** is the longitudinal limits (in degrees) of the area eligible for longitude labeling. The western limit must be the first value. If the value -9999.0 (default) is entered, the limits specified in the image's header will be used. If GRID="Y", the grid's LON_RNGE values will be the default values. LON_RG/LON_DLT specify which longitudes are to be labeled whereas LON_LT_R/LON_LT_D specify where along these longitudes the labels are to fall. Geocoordinate points (lat/lon intersections) whose longitudes will be labeled must meet several criteria. First, they must fall within the geographic area specified by LON_RG and LON_LT_R. Second, the corresponding image points (pixel/line) must be located within the WINDOW limits. Third, they must fall on longitudes which are zero or more LON_DLT intervals from the western LON_RG. Finally, they must fall on latitudes which are zero or more LON_LT_D intervals from the southern LON_LT_R. See these other label parameters for additional information.

(17) **LON_DLT** is the spacing (in degrees) between longitudes within LON_RG which are eligible for longitude labeling. If the value -9999.0 (default) is entered, a reasonable value will be used based on the longitudinal limits. If GRID= "Y", the grid's LON_DELT value will be the default value. See the LON_RG label parameter for information on which longitudes are labeled.

(18) **LON_LT_R** is the latitudinal limits (in degrees) of the area eligible for longitude labeling. If the value -9999.0 (default) is entered, LAT_RG will be used. See the LON_RG label parameter for information on which longitudes are labeled.

(19) **LON_LT_D** is the spacing (in degrees) between latitudes within LON_LT_R at which to label longitudes. If the default value -9999.0 is entered, the spacing will be larger than LON_LT_R; i.e., only one label per longitude will be used along the southern LON_LT_R latitude. See the LON_RG label parameter for additional information on which longitudes are labeled.

(20) **LON_OFFS** are the offsets (in pixels) to apply to longitude labels which would appear at the left, right, top, and bottom edges of the display, respectively. The offsets may be positive or negative. The left and right offsets are used to obtain a linear equation of the horizontal offset for labels as a function of the pixel (horizontal) location. Likewise, the top and bottom offsets are used to obtain a linear equation of the vertical offset for labels as a function of

the line (vertical) location. These offsets may be used to avoid labels overlapping or to place them off the longitudes to which they correspond. If the offsets are all zero (default), the center of each label will be located atop the image point of the geographic location to which it corresponds.

Latitude Label Parameters:

(21) **LAT_RG** is the latitudinal limits (in degrees) of the area eligible for latitude labeling. If the value -9999.0 (default) is entered, the limits specified in the image's header will be used. If GRID="Y", the grid's LAT_RG values will be the default values. LAT_RG/LAT_DLT specify which latitudes are to be labeled whereas LAT_LN_R/LAT_LN_D specify where along these latitudes the labels are to fall. Geocoordinate points (lat/lon intersections) whose latitudes will be labeled must meet several criteria. First, they must fall within the geographic area specified by LAT_RG and LAT_LN_R. Second, the corresponding image points (pixel/line) must be located within the WINDOW limits. Third, they must fall on latitudes which are zero or more LAT_DLT intervals from the southern LAT_RG. Finally, they must fall on longitudes which are zero or more LAT_LN_D intervals from the western LAT_LN_R. See these other label parameters for additional information.

(22) **LAT_DLT** is the spacing (in degrees) between latitudes within LAT_RG which are eligible for latitude labeling. If the value -9999.0 (default) is entered, a reasonable value will be used based on the latitudinal limits. If GRID="Y", the grid's LAT_DELT value will be the default value. See the LAT_RG label parameter for information on which latitudes are labeled.

(23) **LAT_LN_R** are the longitudinal limits (in degrees) of the area eligible for latitude labeling. The western limit must be the first value. If the value -9999.0 (default) is entered, LON_RG will be used. See the LAT_RG label parameter for information on which longitudes are labeled.

(24) **LAT_LN_D** is the spacing (in degrees) between longitudes within LAT_LN_R at which to label latitudes. If the default value -9999.0 is entered, the spacing will be larger than LAT_LN_R; i.e., only one label per latitude will be used along the western LAT_LN_R longitude. See the LAT_RG label parameter for additional information on which longitudes are labeled.

(25) **LAT_OFFS** are the offsets (in pixels) to apply to latitude labels which would appear at the left, right, top, and bottom edges of the display, respectively. Equivalent to LON_OFFS.

General Label Parameters:

(26) **TXT_MODE** is the flag of the text mode selection in HALO88 which is to be used for labeling. If the value 1 is entered, the dot text mode will be used. Otherwise, the stroke text mode will be used.

(27) **DECIMAL** is the number of decimal places to appear in each label. The values of each longitude or latitude will be rounded off to this number of decimal places. If zero

(default) is specified, the values will be rounded to the nearest integer and the decimal point will not appear. However, the number of decimal places will be increased if the interval spacing (degrees) is too small to differentiate the label values using the specified DECIMAL or if DECIMAL=0 and the spacing is not a whole number.

(28) **G_PAL** is the value of the graphics palette to use for the labels and border. If the value is negative, any graphics already created with that graphics palette will first be deleted. Graphics palette 1 (positive) is the default value. If GRID="Y", the grid's G_PAL value will be the default value.

(29) **WINDOW** defines a rectangular view area which may be all or part of the entire image display area. Only geocoordinate points (latitude/longitude intersections) falling within this view area will be labeled. The WINDOW values represent the start pixel, end pixel, start line, and end line numbers of the view area in that order. The maximum display area is 512 pixels wide by 512 or 480 lines (depending on the MVP-AT configuration) high and is used by default. If GRID="Y", the grid's WINDOW values will be the default values.

(30) **BORDER** determines if a border is to be drawn around the view area. If "Y", straight lines will be drawn around the view area defined by WINDOW label parameter. "N" is the default value. Note that labels for geocoordinate points located near the edges of WINDOW may overlap the border.

Label Character (in Dot Text Mode) Parameters:

(31) **DOT_FONT** is the font file selection for dot text. There are seven available fonts to be selected.

1. HALO 88 default	2. HALO001.FNT	3. HALO002.FNT
4. HALO010.FNT	5. HALO011.FNT	6. HALO012.FNT
7. HALO013.FNT		

The font files (except the default) must exist under the HALO88 directory specified in the program SPKSETUP. For font definitions of those files, please check the appendix in this guide.

(32) **DOT_H** is the height in pixels of the dot text. This should always be a multiple of 8.

(33) **DOT_W** is the width in pixels of the dot text. This should always be a multiple of 8.

(34) **DOT_ANG** is the direction in degree of the dot text to be displayed. Only 0, 90, 180 and 270 are valid inputs.

(35) **GB_PAL** is the graphics palette to be used for the background of the dot text, only 0 to 7 are valid inputs.

Label Character (in Stroke Text Mode) Parameters:

(36) **STRK_FNT** specifies the font file selection for stroke text. There are 19 available fonts to be selected.

1. HALO102.FNT	2. HALO103.FNT	3. HALO104.FNT
4. HALO105.FNT	5. HALO106.FNT	6. HALO107.FNT
7. HALO108.FNT	8. HALO109.FNT	9. HALO111.FNT
10. HALO115.FNT	11. HALO201.FNT	12. HALO202.FNT
13. HALO203.FNT	14. HALO204.FNT	15. HALO205.FNT
16. HALO206.FNT	17. HALO207.FNT	18. HALO208.FNT

19. HALO209.FNT
The font files (except the default) must exist under the HALO88 directory specified in the program SPKSETUP. For font definitions of those files, please check the appendix in this guide.

(37) **STRK_H** is the height in pixels of the stroke text.
(38) **STRK_ANG** is the direction in degree of the stroke text to be displayed.
(39) **STRK_ASP** is the aspect ratio of the stroke text.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: HDRCVT
DATE: 10/28/91
MENU: VAXTOPC

DESCRIPTION: The program HDRCVT is used to convert the header block of an image file from the VAX SEAPAK format into the PC-SEAPAK format. It allows the SEAPAK image files created on the VAX to be used in the PC. The header in all SEAPAK image files transferred from the VAX have to be converted to PC format using this program since the real data under the VAX and PC use different data representation methods.

PARAMETER:
(1) **IFILE** is the input image file name(s) whose header(s) is to be converted. The wild card characters "*" and "?" may be used to specify a group of files. "?" is used to replace a single character and "*" is used to replace multiple characters.
(2) **CTLFILE** is the new control point file name, under PC-SEAPAK and related to the image file IFILE, to be updated into the header block.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: HILOW
DATE: 10/28/91
MENU: IMGFILE

DESCRIPTION: This program generates an output image file which has the minimum or the maximum gray level value for each pixel from a set of input image files. The output file has a blank header block. The gray level ranges can be specified to restrict the input images to be examined.

PARAMETERS:

(1) **IMGFILS** are the input image file names to be processed. Up to 36 file names may be entered. However, since the wild card (* or ?) file format is supported, up to 300 image files can be processed. Note that if there is only one file entered for IMGFILS, the program will assume it is a text file and read the input image files from this file. Note that all the image files should have a header block.

(2) **O_FIL** is the output file name to be generated. A blank header block will be added at the beginning of this file.

(3) **MIN_MAX** decides whether the minimum or the maximum values from the input image files will be used to generate the output file. A "0" uses the minimum and a "1" uses the maximum to generate the output file.

(4) **RANGE** is a range of gray level values to be used with the parameter IN_OUT to decide the pixels in the input image files to be processed. If IN_OUT is "1", then only pixels with gray level within (inclusive) RANGE will be processed. If IN_OUT is "0", then only pixels outside (exclusive) RANGE will be processed.

(5) **INVALID** is a gray level value to be assigned for the output pixels, if the corresponding pixels of all the input image files fail the RANGE and IN_OUT test. This parameter will not be used, if RANGE is "0" and "255". Otherwise, the value for this parameter has to be outside (exclusive) of RANGE, if IN_OUT is "1", or inside (inclusive) of RANGE, if IN_OUT is "0".

(6) **IN_OUT** is a flag to be used with parameter RANGE to decide the pixels in the input image files to be processed. If IN_OUT is "1", then only pixels with gray level within (inclusive) RANGE will be processed. If IN_OUT is "0", then only pixels outside (exclusive) RANGE will be processed.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: HIST
DATE: 10/28/91
MENU: STAT1

DESCRIPTION: This program enables the user to generate regular, cumulative and aggregated histograms on gray level, sea surface temperature (SST), or pigment values over the full image or a user defined rectangular area or blotch areas (including a line). Additional options include the capability to specify the histogram range, to change the overlay graphics palette, to write the histogram data out to a file or to the printer, to load a blotch area from a file created using the program BLOTCH and to display the current cursor position in pixel and line coordinates. Note that the frame buffer 1 (if the overlay frame buffer is 0) will be destroyed if the user is doing a histogram on a rectangular area or blotch areas.

PARAMETERS:
There are no initial input parameters.

DYNAMIC PARAMETERS:
I. Used for changing the histogram gray level range.
(1) **GLV1** is the minimum gray level (0-255) for the histogram to be displayed.
(2) **GLV2** is the maximum gray level (0-255) for the histogram to be displayed.
II. Used for changing the histogram SST range.
(1) **SEATEMP1** is the minimum SST (0.0 - 32.0 deg. C) for the histogram to be displayed.
(2) **SEATEMP2** is the maximum SST (0.0 - 32.0 deg. C) for the histogram to be displayed.
III. Used for changing the pigment concentration range.
(1) **PIGMNT1** is the minimum pigment concentration (0.0 - 45.7088 mg/m3) for the histogram to be displayed. To exclude land from being histogrammed, enter a value 0.04093 or higher.
(2) **PIGMNT2** is the maximum pigment concentration (PIGMNT1 - 45.7088 mg/m3) for the histogram to be displayed.
IV. Used for changing the breakpoints for aggregated histogram.
(1) **BRK_PNT** is the gray level breakpoints (up to 29) for aggregation. The aggregation is done as follows:
   a. AGG[1] = sum up histogram data from BRK_PNT(1) to BRK_PNT(2)
   b. AGG[I] = sum up histogram data from BRK_PNT(I)+1 to BRK_PNT(I+1) for I=2 to (max no. of breakpoints - 1)
V. Used for outputting histogram data to file or printer.
(1) **O_FILE** is the output file name, or LPT1 for printer, where the data of a specified histogram will be saved or sent to.
(2) **H_MODE** is the histogram type to be outputted. Enter 1 for regular histogram, 2 for cumulative histogram or 3 for aggregated histogram.

(3) **D_MODE** is the histogram data mode to be outputted. Enter 1 for gray level data, 2 for SST data or 3 for pigment concentration data. This parameter is valid only when the histogram type is regular or cumulative.
(4) **CHART_FG** is a flag to specify whether to generate the histogram bar chart in text mode on the output. Enter 0 for No, or 1 for Yes.

VI. Used for loading blotch areas from a file.
(1) **BLOFIL1** is the input file of overlay graphics which contains 512x512 bytes data with values between 0 and 7 and is to be loaded into the overlay frame buffer 0.

VII. Used for changing the graphics palettes.
(1) **HIST_PAL** is the graphics palette (1 to 7) to be used for drawing the histogram.
(2) **LB_PAL** is the graphics palette (1 to 7) to be used for drawing the axes and the tick marks for the histogram.
(3) **TX_PAL** is the graphics palette (1 to 7) to be used for writing the text title of the histogram in the X-axis.
(4) **CUR_PAL** is the graphics palette (1 to 7) to be used for displaying the cursor.

FUNCTION KEY DEFINITIONS:

ESC: Exits the program HIST.

F1: Generates the regular histogram. Depending on current status, the histogram generated can be any combination of the three sampling modes (full image, rectangular or blotch areas) and the three image data modes (gray level, SST and pigment). It is also restricted by the current range set for the three image data modes.

F2: Generates the cumulative histogram. Depending on current status, the histogram generated can be any combination of the three sampling modes (full image, rectangular or blotch areas) and the three image data modes (gray level, SST and pigment).

F3: Generates the aggregated histogram. Depending on current status, the histogram generated can be one of the three sampling modes (full image, rectangular or blotch areas) but only the gray level data can be aggregated with up to 29 intervals.

F4: Allows the user to change current sampling mode by looping through the full image mode, the rectangular mode and the blotch areas mode.

F5: Allows the user to change current image data mode by looping through the gray level, the SST and the pigment concentration modes.

F6: Changes current graphics palette by increasing the palette number by one. If the value is greater than seven, it will be reset to 1. Note that when displaying the histogram in the blotch area sampling mode, only those blotch areas created with the current graphics palette will be sampled.

F7: Allows the user to change the histogram range along the X-axis. The user is prompted for two values in units of gray level, SST or pigment concentration depending on current

status. These two values correspond to the lower and upper bounds of the histogram to be displayed. To exclude the land from the histogram in the pigment mode, enter a lower bound value of 0.04093 or higher.

F8: Allows the user to set up the breakpoints for the aggregated histogram. Up to 29 integer values in the range (0, 255) may be entered.

F9: Allows the user to define a rectangular area to be sampled for the histogram. Once the key is pressed, the overlay graphics will be cleared and only the rectangle created previously, if any, and the cursor will be displayed. The user can move the cursor to any position and press the left mouse button to define or redefine the first corner point of the rectangle. Once the first corner point is defined, the rectangular will be created automatically when the mouse is moved. Depending on the mouse movement, the rectangle can be any size. To finish, press the right mouse button and the rectangle currently displayed will be used for sampling if the sampling mode is rectangular.

F10: Allows the user to define the blotch areas to be sampled for a histogram. Once the key is pressed, a new function key set will be defined and the blotch areas created previously, if any, and the cursor will be displayed. At this time, the user can move the cursor around and press F1 to define a new vertex or F2 to erase last vertex. As many as 500 vertices can be defined for each blotch area and up to 10 blotch areas can be defined. Key F3 is used to close the blotch area (connect the last vertex and the first vertex and fill the area with the color defined in current graphics palette). Key F4 is used to erase a blotch area. This can only be done when the cursor is inside the blotch area. Key F5 is used to change the current graphics palette. After the blotch areas are created, the ESC key has to be used to return to the main function key set.

ALT F5: Allows the user to load the blotch areas from a file created previously using the program BLOTCH.

ALT F6: Allows the user to change the graphics palettes used to draw the histogram, the axes, the tick marks and the text labels.

ALT F8: Enables the cursor position in pixel and line coordinates to be displayed.

ALT F9: Allows user to write the histogram data to a file or to the printer. The user will be prompted for the file name, the histogram mode (regular, cumulative or aggregated), the image data mode (gray level, SST or pigment concentration), the flag to output the histogram in a text plot mode and the minimum and maximum gray level ranges.

ALT F1: Toggles the function key menu display on/off.

MOUSE RIGHT BUTTON: Same as ALT F1.

PROGRAM NAME: IMAGE
DATE: 10/28/91
MENU: FRMBUF

DESCRIPTION: IMAGE loads a disk-resident image into a specified frame buffer (0 to 3). The image to be displayed is 512x512 bytes (one byte per pixel) but it may have a varying number of 512-byte header blocks. This number can be specified and thus prevents part of the header from being displayed as part of the image. It is also possible to specify a frame buffer in which to load the image but not to display it on the image display monitor.

PARAMETERS:
(1) **IMGFILE** is the name of the disk file containing the image to drop. This must be a standard 512x512x8 bit image with a known number of header blocks.
(2) **FRMBUF** is the index number (0-3) of the frame buffer to receive the image.
(3) **HEADNO** specifies the number of 512-byte header blocks in the image. This number of blocks will be skipped before reading the image data.
(4) **YNIMG** is a flag indicating whether to display the dropped image (1) or leave on the current frame buffer being displayed (0).

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: IMAGSAV
DATE: 10/28/91
MENU: FRMBUF

DESCRIPTION: The program IMAGSAV saves a 512x512x8 bit image from a frame buffer into a file. The file can be saved with or without a PC-SEAPAK header.

PARAMETERS:
(1) **OFIL** is the output file name to save the image.
(2) **FRMBUF** is the index number (0 to 3) of the frame buffer to be saved into a file.
(3) **HDRFILE** is the image file name whose header will be copied to the output image file. If this parameter is blank (default) then no header block will be created in the output file. If the file specified in this parameter does not exist or has error in opening or reading it, a blank header block will be created in the output file.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: IMATCH
DATE: 10/28/91
MENU: SOFTFCT

DESCRIPTION: This program tests gray level values in the file IFIL1. If a pixel has a value whithin the range IN_RANGE, the corresponding pixel in the file IFIL2 is reset to the value OUT_GRAY in the file OFIL. All remaining pixels in OFIL retain the original values in IFIL2. One application of this program is to flag invalid pigment and diffuse attenuation coefficient values derived from invalid water radiance values as determined from the level 2 images. This program can also be used to burn an overlay graphics file into a regular image file.

PARAMETERS:

(1) **IFIL1** is the name of the image disk file whose pixel gray levels will be tested for IN_RANGE values. Pixels that test positive will cause the corresponding IFIL2 pixels to be set equal to OUT_GRAY and output to OFIL. Note that you can specify the same file for IFIL1 and IFIL2 in order to reset the values of pixels in that file that fall within IN_RANGE to the OUT_GRAY value.

(2) **IFIL2** is the name of the image disk file whose pixels are to be set equal to OUT_GRAY in OFIL. IFIL2 pixels corresponding to IFIL1 pixels that test positive for IN_RANGE values will be set to OUT_GRAY in OFIL. Note that you can specify the same file for IFIL1 and IFIL2 in order to reset the values of pixels in that file that fall within IN_RANGE to the OUT_GRAY value.

(3) **OFIL** is the name of the output file to be created. The OFIL will have the same header block as IFIL2 if HDRBLK2 is 1.

(4) **IN_RANGE** is the gray level range (inclusive) to use for testing IFIL1 pixels. Pixels that test positive will cause the corresponding IFIL2 pixels to be set equal to OUT_GRAY in OFIL.

(5) **OUT_GRAY** is the gray level value at which to set IFIL2 pixels for the output file OFIL, if the values of corresponding IFIL1 pixels fall within IN_RANGE.

(6) **HDRBLK1** is the number of blocks (512 bytes/block) in the header of IFIL1. This header, if any, will be skipped before processing the gray values.

(7) **HDRBLK2** is the number of block (512 bytes/block) in the header of IFIL2. This header, if any, will be used as the header of the output file OFIL.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: IMG2FCT
DATE: 10/28/91
MENU: HARDFCT

DESCRIPTION: This program performs a specified operation between two images stored in frame buffers. Any of seven functions may be selected: addition, subtraction, minimum, maximum, logical AND, logical OR, and logical XOR.

PARAMETERS:

(1) **SBUF1** and SBUF2 are the frame buffers (0-5) on which the function FUNCT will be performed. Note that, when FUNCT is MIN or MAX, SBUF1 and SBUF2 cannot be part of the 16-bit frame buffer to which DBUF belongs. (16-Bit frame buffers 4 and 5 are used to refer to buffers of the image memory configured as two 16-bit buffers where 4 is equivalent to the regular 8-bit buffers 0/1 and 5 is equivalent to buffers 2/3.)

(2) **SBUF2** and SBUF1 are the frame buffers (0-5) on which the function FUNCT will be performed. Note that, if FUNCT is MIN or MAX, SBUF1 and SBUF2 cannot be part of the 16-bit frame buffer to which DBUF belongs. (16-Bit frame buffers 4 and 5 are used to refer to buffers of the image memory configured as two 16-bit buffers where 4 is equivalent to the regular 8-bit buffers 0/1 and 5 is equivalent to buffers 2/3.)

(3) **DBUF** is the frame buffer to receive the resulting image. Any frame buffer number from 0 to 5 may be specified expect that, if FUNCT is MIN or MAX, only 8-bit buffers 0 to 3 may be entered. (DBUF values of 4 and 5 are used to refer to frame buffers of the image memory configured as two 16-bit buffers where 4 is equivalent to the regular 8-bit buffers 0/1 and 5 is equivalent to buffers 2/3.) The results will be displayed on the monitor. If DBUF is 4 or 5, only the low-byte frame buffer (i.e., the regular buffers 0 or 2) of each will be displayed.

(4) **FUNCT** specifies the operation to be performed on the frame buffers SBUF1 and SBUF2:

```
0 (ADD) - SBUF1 pixel value + SBUF2 pixel value
1 (SUB) - SBUF1 pixel value - SBUF2 pixel value
2 (MIN) - min(SBUF1 pixel value, SBUF2 pixel value)
3 (MAX) - max(SBUF1 pixel value, SBUF2 pixel value)
4 (AND) - (SBUF1 pixel bits) AND (SBUF2 pixel bits)
5 (XOR) - (SBUF1 pixel bits) XOR (SBUF2 pixel bits)
6 (OR)  - (SBUF1 pixel bits) OR (SBUF2 pixel bits)
```

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: IMGEDIT
DATE: 10/28/91
MENU: FRMBUF

DESCRIPTION: This program provides capabilities for cutting and shifting a region of interest (ROI) on a displayed image. These capabilities may be used to generate composite (mosaic) images. The ROI may be specified as being inside or outside the blotch (colored) areas of the current graphics palette. Function keys are provided to define blotch areas and to manipulate the frame buffers and graphics palettes.

PARAMETERS:
There are no parameters.

DYNAMIC PARAMETERS:
I. Parameters for cutting and shifting an image.

(1) **CUT_IMG** is used as a flag. If "YES", the areas outside of the region of interest (ROI) will be set to the CUT_VAL gray value in the output frame buffer O_BUF. However, if O_BUF is occupied and CUT_IMG="YES", you will be prompted for the option to insert (after any shifting) the ROI into the image residing in O_BUF. You may use this option to compose a mosaic image in O_BUF. When O_BUF is occupied, you will also have the option to cancel the cut/shift function or to clear O_BUF before dropping the edited image. If "NO", the areas outside the ROI will retain the gray values of the corresponding pixels in the input channel. In this case, these areas will appear as in the input channel since they will also not be shifted. (Only the ROI can be shifted.) The ROI may be defined as being inside or outside the blotch (colored) areas of the current graphics palette (see IN_OUT).

(2) **CUT_VAL** is the gray level value (0 to 255) to set the areas outside the region of interest (ROI) if CUT_IMG = "YES".

(3) **SHIFT** is used to select the method of shifting. Enter "X/Y" if you will specify the number of pixels to shift the region of interest (ROI) of the input image using DELT_XY. Otherwise, if "MANUAL" is specified, you will be able to use the mouse to define the extent of the shift. If SHIFT="X/Y" and DELT_XY=(0,0), no shifting will occur. The ROI may be defined as being inside or outside the blotch (colored) areas of the current graphics plane (see IN_OUT).

(4) **DELT_XY** is used to specify the number of pixels to shift the region of interest (ROI) of the input image when SHIFT="X/Y". If SHIFT="X/Y" and DELT_XY=(0,0), no shifting will occur.

(5) **WRAP_XY** is used, if WRAP(1)="YES" and WRAP(2)="YES", to indicate that wrapping of the region of interest (ROI) is desired for the X (pixel) and Y (line) directions during shifting. Note that when wrapping is not requested, the direction of the shift is important; for example, +256 is not the same as -256.

(6) **I_BUF** is the frame buffer (1 to 3) containing the image to be edited.

(7) **O_BUF** is the frame buffer (1 to 14) to receive the edited image. If O_BUF is occupied and cutting is requested (CUT_IMG="YES"), you will be prompted for the option to insert (after any shifting) the region of interest (ROI) into the image residing in O_BUF. You may use this option to compose a mosaic image in O_BUF. When O_BUF is occupied, you will also have the option to cancel the cut/shift function or to clear O_BUF before dropping the edited image.

(8) **DISP_OUT** is a flag for displaying (when "YES" is entered) the image generated for O_BUF after processing the cut/shift function.

(9) **IN_OUT** is used to select the inside or outside of blotch (colored) areas for region of interest (ROI). Enter "IN" if ROI is within the blotch areas of the current graphics palette; enter "OUT" if it is outside those areas. The ROI is that portion of the I_BUF image which will not be cut if CUT_IMG="YES" and will be shifted if a shift has been specified.

II. Parameters for dropping a new image.

(1) **IMGFILE** is the name of the disk file containing the image to drop. This must be a standard 512x512x8 bit image with a known number of header blocks.

(2) **FRMBUF** is the index number (1-3) of the frame buffer to receive the image.

(3) **HEADNO** specifies the number of 512-byte header blocks in the image. This number of blocks will be skipped before reading the image data.

(4) **YNIMG** is a flag indicating whether to display the dropped image (1) or leave on the current frame buffer being displayed (0).

III. Parameters for saving a displayed image into a file.

(1) **OFIL** is the output file name to save the image.

(2) **FRMBUF** is the index number (0 to 3) of the frame buffer to be saved into a file.

(3) **HDRFILE** is the image file name whose header will be copied to the output image file. If this parameter is blank (default) then no header block will be created in the output file. If the file specified in this parameter does not exist or has error in opening or reading it, a blank header block will be created in the output file.

IV. Parameters for clearing the graphics palette(s).

(1) **CLR_PAL** is the number of the palette from which to clear overlay graphics. A "-1" is to used to clear all overlay graphics.

V. Parameter for saving blotch graphics into a file.

(1) **BLOFILE** is the file name in which to save all the overlay graphics created in this program. The output file will contain 512x512 bytes of data from the overlay frame buffer 0 without any header blocks.

VI. Parameter for restoring blotch graphics from a file.

(1) **BLOFIL1** is the file name of the overlay graphics which are to be loaded into overlay frame buffer 0.

FUNCTION KEY DEFINITIONS:

ESC: Exits the program.

F1: Asks the user to enter parameters (see DYNAMIC PARAMETERS I) for cutting and shifting an image.

F2: Allows the user to drop a new image into the frame buffer.

F3: Allows the user to save a displayed image into a file.

F4: Displays current cursor position (1 relative) and the gray level value of the displayed image at that point.

F5: Switches the displayed image between I_BUF and O_BUF (see DYNAMIC PARAMETERS I).

F6: Switches the displayed image between frame buffers 1, 2, and 3.

F7: Turns all graphics palettes on/off.

F8: Turns the displayed image on/off.

F9: Increases the current graphics palette by 1 or resets it to 1 if the value is greater than 7.

F10: Clears all the overlay graphics or a specified graphics palette. The parameter CLR_PAL will be requested.

ALT F1: Toggles function key menu display on/off.

ALT F5: Allows the user to define the blotch areas to be used for cutting and shifting of the image data. Once the key is pressed, a new function key set will be defined and the blotch areas created previously, if any, and the cursor will be displayed. At this time, the user can move the cursor around and press F1 (or the mouse left button) to define a new vertex or F2 to erase the last vertex. As many as 500 vertices can be defined for each blotch area and up to 10 blotch areas can be defined. Key F3 is used to close the blotch area (connect the last and first vertices and fill the area with the color defined for the current graphics palette). Key F4 is used to erase a blotch area; this can only be done when the cursor is inside the blotch area. After the blotch areas are created or modified, the ESC key has to be used to return to the main function key set.

ALT F6: Redisplays the blotch areas defined by ALT F5.

ALT F7: Allows the user to save the current overlay graphics or blotches (in frame buffer 0) into a file.

ALT F8: Allows the user to restore overlay graphics or blotches from a file into frame buffer 0.

MOUSE RIGHT BUTTON - Toggles function key menu display on/off.

PROGRAM NAME: IMGFCT
DATE: 10/28/91
MENU: HARDFCT

DESCRIPTION: This program performs a specified operation on an image in an image buffer and a constant. Any of twelve functions may be selected: addition, subtraction, minimum, maximum, division, multiplication, logical AND, logical OR, logical XOR, unary NOT, absolute, and clipping.

PARAMETERS:

(1) **SBUF** is the frame buffer (0-5) on which the function FUNCT will be performed. (16-Bit frame buffers 4 and 5 are used to refer to buffers of the image memory configured as two 16-bit buffers where 4 is equivalent to the regular 8-bit buffers 0/1 and 5 is equivalent to buffers 2/3.)

(2) **DBUF** is the frame buffer (0-5) to receive the resulting image. (DBUF values of 4 and 5 are used to refer to frame buffers of the image memory configured as two 16-bit buffers where 4 is equivalent to the regular 8-bit buffers 0/1 and 5 is equivalent to buffers 2/3.) DBUF 4 or 5 should be used unless the user is certain that the results do not exceed 8 bits (signed). The results will be displayed on the monitor. If DBUF is 4 or 5, only the low-byte frame buffer (i.e., the regular buffers 0 or 2) of each will be displayed.

(3) **FUNCT** specifies the operation to be performed on the frame buffer SBUF:

```
 0 (ADD)   - pixel value + VALUE
 1 (SUB)   - pixel value - VALUE
 2 (MIN)   - min(pixel value, VALUE)
 3 (MAX)   - max(pixel value, VALUE)
 4 (DIV)   - pixel value / VALUE
 5 (MUL)   - pixel value * VALUE
 6 (AND)   - bitwise logical AND (pixel value, VALUE)
 7 (XOR)   - bitwise logical Exclusive OR (pixel value, VALUE)
 8 (OR)    - bitwise logical OR (pixel value, VALUE)
 9 (NOT)   - unary NOT (pixel value)
10 (ABS)   - abs(pixel value)
11 (CLIP)  - min(pixel value, 255)
```

(4) **VALUE** is the constant to be used in the operation specified by FUNCT with all pixel values in the frame buffer SBUF. When performing the NOT, ABS, or CLIP operation, this parameter is ignored. Values of -2**15 to 2**15 may be entered.

(5) **PREC** is the number of bits upon which the operation FUNCT will be performed. This parameter is used only when FUNCT is MUL or DIV. Values of 1 to 12 may be entered.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: <u>IMGLST</u>
DATE: <u>10/28/91</u>
MENU: <u>FRMBUF</u>

DESCRIPTION: IMGLST lists the file names or comments of the images currently in the MVP-AT frame buffers. Whenever the image in any of the frame buffers is changed by any program in PC-SEAPAK, a catalog file will be updated to keep the recent information related to that frame buffer. The catalog file is stored under the SEAPAK directory with the name IMGLST.DAT.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: IMGPRINT
DATE: 10/28/91
MENU: PAINTJET

DESCRIPTION: This program generates a color hard-copy of an image on the PaintJet printer. The image to be printed can be stored in a file or in a frame buffer on the MVP-AT. An optional graphics overlay from another file or frame buffer can also be printed with the image. The colors used by the PaintJet to print an image are defined in the file specified in the parameter PAL_FIL which is created using program PJTCOL. There are up to 16 colors for the displayed image and 7 colors for the overlay graphics. Sixteen default break points for gray level values will be displayed and the user will be allowed to change these before the image is printed. Each interval has a color corresponding to one of the 16 colors defined by PAL_FIL. This program also allows the user to save the output into a file by specifying a file name in the parameter PRT_FILE instead of the default LPT1 line printer. The file generated here may be sent to the PaintJet printer by using the DOS command "COPY /B FILENAME.EXT LPT1." Note that the DOS command "PRINT" cannot be used to print the output file with the PaintJet printer.

PARAMETERS:

(1) **IMAGE** is the source of the image to be printed. A blank value for this parameter means no image will be printed. If a value of "0", "1", "2" or "3" is entered, the image in the frame buffer of the number entered will be used, otherwise it assumes an image from a file will be used.

(2) **OVERLAY** is the source of the overlay graphics to be printed with the image. A blank value for this parameter means no overlay will be printed. A value "0", "1" , "2" or "3" will use the frame buffer with the number entered for the overlay graphics. Otherwise, an overlay graphics from a file will be used.

(3) **PRT_FIL** is the output file specification. The default is "LPT1" which means the PaintJet printer is in port 1 on the PC. Otherwise, a disk file name is assumed and all the output will be sent to that file. The output file created here can be sent to the PaintJet printer with the DOS command "COPY /B FILENAME.EXT LPT1".

(4) **PAR_FIL** is a file which contains the specification of colors to generate the hard copy of an image on the PaintJet printer. This file must be created using program PJTCOL.

(5) **LUT_FIL** is used to determine the default gray level break point values used in parameter GRAY_LEVEL. If this parameter is blank and the image is from a file, 16 equally spaced gray level values from 0 to 255 will be used for the default 16 break points. If this parameter is blank and the image is from a frame buffer, the look-up table of that frame buffer will be used to determine the default break points. Other-

wise, if this parameter is not blank, the program will use the look-up tables in the file to decide the default break points. The file entered here must be created using program TABSAV or PAINT.

(6) **S_LINE** is the starting line of the image to be printed. A value between 1 and 512 should be entered.

(7) **E_LINE** is the ending line of the image to be printed. A value between S_LINE and 512 should be entered.

(8) **H_OFFSET** is the horizontal paper offset in inches. Each row of the image will be shifted right H_OFFSET inches when printing. Only values between 0.0 to 2.2 are valid input.

(9) **V_OFFSET** is the vertical paper offset in inches. The paper will scroll up V_OFFSET inches before printing. Only values between 0.0 and 5.0 are valid input.

(11) **IMG_HDR** is the number of header blocks in the image file. This is used only when a file name is entered for the parameter IMAGE.

(12) **OVI_HDR** is the number of header blocks in the overlay graphics file. This is used only when a file name is entered for the parameter OVERLAY.

(13) **GRAY_LEVEL** is the array of the gray level values for the break points. This parameter will be requested only when IMAGE is not blank. There are 17 values available for this array which may define up to 16 break intervals for selecting the colors of the PaintJet printer. The values should be entered in order. Note that all the gray level values within 0 to 255 which are not covered in any of the break intervals will be assigned to the break interval 16. For example, if the GRAY_LEVEL array is (0, 10, 40, 100, 150, 220) then break intervals 1 to 4 will contain (0, 10), (11, 40), (41, 100), (101, 150) and (151, 220), and all other gray level values (221, 255) will be assigned to the break interval 16.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: IMGXRT
DATE: 10/28/91
MENU: FRMBUF

DESCRIPTION: This program allows the user to extract a PC-SEAPAK image from an input file into an image frame buffer or a disk file. The input file can be in PC-SEAPAK or non-PC-SEAPAK image file format. The program can handle any input image file having a fixed number of pixels on each line with integer (1, 2 or 4 bytes) or real (4 bytes) data. The output PC-SEAPAK image file will contain 512 bytes of blank header followed by 512 (pixels) x 512 (lines) bytes of data.

PARAMETERS:

(1) **I_FIL** is the input file name. This file must be binary consisting of 1 (integer), 2 (integer) or 4 (integer or real) bytes of data per pixel.

(2) **FRM_FIL** is the frame buffer number or the file name for the output image. An entry of 0 to 3 indicates that the output image is to be displayed in that frame buffer; any other entry will be used as the name of a file to create for saving the output image.

(3) **WIDTH** is the number of pixels contained on each line of the input image files. This value cannot exceed 2048.

(4) **BYTES** specifies the number of bytes per pixel of the input image files. An entry of 1, 2, or 4 indicates the number of bytes per integer value, whereas -4 indicates real-valued (4 byte) data.

(5) **SLOPE** is the slope for converting the input values to the output values to be displayed or saved:

```
Output = min(max((Input - INTCPT) * SLOPE), 0), 255)
```

At the end, the program will display the minimum and maximum values of the area defined by WINDOW. These values may be used to select SLOPE and INTCPT for a rerun.

(6) **INTCPT** is the intercept for converting the input values to the output values to be displayed or saved:

```
Output = min(max(Input - INTCPT) * SLOPE), 0), 255)
```

At the end, the program will display the minimum and maximum values of the area defined by WINDOW. These values may be used to select SLOPE and INTCPT for a rerun.

(7) **WINDOW** defines, in conjunction with REDFAC, the area of the input image to use for displaying or generating a PC-SEAPAK image. WINDOW(1) and WINDOW(3) specify the positions of the first and last pixels, respectively, to use from each input image line and WINDOW(2) and WINDOW(4) specify the first and last line numbers, respectively. Note that WINDOW(3) cannot be greater than WIDTH.

(8) **REDFAC** are the pixel and line reduction factors. Positive values indicate reduction by subsampling whereas negative values indicate magnification by pixel replication. For example, an entry of (2,2) will create images half as wide in pixels and half as high in lines as the scene area defined by WINDOW; an entry of (-2,-2) will generate images twice as high and wide. Values of -1, 0, or 1 are equivalent and generate images having a one-to-one correspondence of pixels with the scene defined by WINDOW.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: INIT
DATE: 10/28/91
MENU: INITIAL

DESCRIPTION: INIT initializes the MVP-AT image processor board by setting the board's memory and I/O address as well as the display format. In PC-SEAPAK, the default memory address and I/O address are set to d000 and 300 in hexadecimal and the display format is set to 1:1 pixel aspect ratio with 512 lines in interlaced mode. To change these default setups, one has to use the program SPKSETUP or use edit to modify the values in the file MVPAT.FIG under the SEAPAK directory. This program will also initialize the look-up tables for palettes 0 to 19 to the default setups defined in file PALETTE.PAR under the SEAPAK directory. In addition, it allows the user to optionally clear the frame buffers.

PARAMETERS:

(1) **YORN** is a flag to determine whether the frame buffers are to be cleared. A value 1 will clear all the frame buffers and a value 0 will not clear all the frame buffers.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: L2BOX
DATE: 10/28/91
MENU: CZCSL2

DESCRIPTION: This program is designed to allow the user to roam a scene (displaying any band) and extract level-2 information within a 3x3 box. Bands 1 to 4 must be in the same directory with the same name convention because the program must collect information from these files in order to make the computations. The user can modify any of the level-2 input parameters and recalculate and save the results by using the options provided on the function key menu. This is particularly useful in fine-tuning images to match validation data without generating final level-2 images.

PARAMETERS:

(1) **G_PAL** is the graphics palette (1-7) that will be used to mark the box.

(2) **ILTOPT** specifies the ILT option: If "1", ephemeris data from the ILT record of the level-1 scene will be used. If "0", much of these data will be obtained from the documentation record or calculated by SEAPAK based on the location and time at the start of the scene.

(3) **CORR** is the index of the correction method to use for calculating total radiances:
 1: Use factors and method (time and gain dependent) of R. Evans (Univ. of Miami).
 2: Use correction factors specified by FACTOR.

(4) **ITERATE** determines which atmospheric correction algorithm is to be used. If 0, the standard Gordon et al. (1988) algorithm is applied. If 1, the iteration algorithm of Smith and Wilson (1981) is used. If the iteration algorithm is used, the user should try using Angstrom exponents equal to zero. The program assigns zero values to pixels which do not converge after 10 iterations. When ITERATE=1, an additional output for water radiance at 670 nm will be computed.

(5) **PIGMENT** allows the user to specify one of two pigment algorithms. The first is the "branching" algorithm of Gordon et al. (1983):

```
1. if Lw(550)=<0, then P=46.34456 (saturated); else,
2. if Lw(443)>0.15,
      then  P = A2 * (Lw(443)/Lw(550))**B2,          (A)
      where log10(A2)=0.053 and B2=-1.71;
      if P>=1.5 and Lw(520)>0
         then  P = A4 * (Lw(520)/Lw(550))**B4,       (B)
         where log10(A4)=0.522 and B4=-2.44;
         if P<1.5, then use (A) above;
3. if Lw(443)=<0.15 and Lw(520)>0, then use (B) above;
4. if Lw(443)=<0.15 and Lw(520)=<0,
      then P=46.34456 (saturated);
```

where Lw represents the water-leaving radiance for the band of the specified wavelength (nm) and P is the pigment concentra-

tion in mg/m^3. The second choice is a three-channel algorithm provided by Dennis Clark (see Muller-Karger et al., 1990) which has the form,

```
P = 5.56 * [((Lw(443) + Lw(520))/Lw(550)]**(-2.252)
```

(6) **NORMWAT** determines if the water radiance values are to be normalized (Gordon and Clark, 1981) in which case specify "1". Enter "0" to specify output of subsurface water radiance images. The calculation of the two radiance values are as follows:

```
(1) Obtain the Lw surface water radiance first:
    Lt = Lr + La + t_up * Lw
    t_up = exp(-(tauR/2 + tauOz)/cos(theta)) * ta
    where Lt: total radiance
          Lr: Rayleigh radiance
          La: aerosol radiance
          TauR : Rayleigh optical thickness
          TauOz: ozone optical thickness
          theta: satellite zenith angle
          t_up : diffuse transmission factor (sensor)
          ta   : aerosol transmittance (1.0)
(2) Calculate the subsurface water radiance Lw_ss:
    Lw_ss = Wref * Wref * Lw / (1.0 - Rho)
    where Wref : refractive index of water (1.34)
          Rho  : Fresnel reflectance
(3) Calculate the normalized water radiance [Lw]:
    [Lw] = Lw/cos(theta0)/t_down
    t_down =  exp(- (TauR/2 + TauOz)/cos(theta0))
    where theta0 : solar zenith angle
          t_down : diffuse transmission factor (sun)
```

The [Lw] calculated should be nearly independent of the solar zenith angle. For pigment concentrations less than 0.25 mg/m3, the values for 520nm and 550nm should be about 0.30 and 0.50, respectively.

(7) **OZ_OPT** is used to select the method of obtaining ozone values for the level-2 data calculation. The default value "1" causes the ozone values for the scene center to be read from the TOMS database; a value of "2" indicates that the ozone values specified by the parameter OZONE are to be used; a value of "3" causes the ozone values to be read from the TOMS database for the current cursor location. During program execution, the user may change or toggle the current selection by using the function key F4 or ALT F4.

(8) **OZONE** are the optical thicknesses (in meters) for bands 1 to 4, respectively. If the value "-999" is entered, the values used will be from the PC-TOMS database for the day of the input CZCS scene and for the point nearest to the image center. If the PC-TOMS data point is missing or an error occurs accessing the data, a message to that effect will be displayed on the terminal along with the default values. These default thicknesses are 0.00106, 0.0144, 0.0279, and 0.0125, and are the products of the absorption coefficients

(3.4E-6, 46E-6, 89E-6, and 40E-6) used at the Univ. of Miami and an average amount of 313 Dobson units of ozone.

(9) **ANGEXP** are the Angstrom exponents for bands 1 to 4, respectively. For the Gordon algorithm, the fourth value is not used.

(10) **FACTOR** are the correction factors to use for calculating total radiances of bands 1 to 4, respectively. These will be used only when CORR=2.

FUNCTION KEY DEFINITIONS:

ESC: Exits the program.

F1: Allows the user to mark the box at the current position in the color defined by current graphics palette.

F2: In order to use the "Show" function keys after changing the input parameters (F4), press this button first so that the program recomputes the necessary quantities. The latitude and longitude of the cursor will be output to the terminal.

F3: Prompts the user for the latitude and longitude coordinates to which the cursor will move.

F4: Prompts the user for revised values of any of the input parameters.

F5: Outputs the mean pigment concentrations for bands 1 to 4.

F6: Retrieves the 3x3 array and computes the mean of the total radiances for bands 1 to 4.

F7: Outputs the 3x3 array and computes the mean of the Rayleigh radiances for bands 1 to 4.

F8: Outputs the 3x3 array and computes the mean of the aerosol radiances for bands 1 to 4.

F9: Outputs the mean water radiances for bands 1 to 4.

F10: Allows the user to list (type) a file previously saved using ALT F10.

ALT F1: Toggles the function key menu display on/off.

ALT F4: Toggles the method of obtaining ozone values for the level-2 data calculation. (See parameter OZ_OPT.)

ALT F10: Generates a text file of the arrays and mean values of all quantities and of all input parameters.

MOUSE RIGHT BUTTON: Toggles the function key menu display on/off.

PROGRAM NAME: L2CON
DATE: 10/28/91
MENU: L2PROD

DESCRIPTION: This program converts a pigment image into a linearly scaled image, or from a linearly scaled image into a pigment image or another linearly scaled image. The pigment and gray level conversion of a pigment image is based on the University of Miami DSP system equations.

PIG = 10.0 ** (0.012 * GRAY - 1.4) (1)

and

GRAY = (log10(PIG) + 1.4) / 0.012 (2)

For linearly scaled image, the equations

DATA = SLOPE * GRAY + INTERCEPT (3)

and

GRAY = (DATA - INTERCEPT) / SLOPE (4)

are used separately for the input and output conversions. In the third equation, the SLOPE and INTERCEPT are extracted from the image header if the input parameter FACTOR is less or equal 0, otherwise (FACTOR greater than 0), the SLOPE = (1/FACTOR) and INTERCEPT = 0. In the fourth equation, the INTERCEPT = MIN_DATA and SLOPE = (MAX_DATA - MIN_DATA) / 255, where the MIN_DATA and MAX_DATA are the minimum and maximum data values of the whole converted image.
The program uses equation (1) or (3), depending on the input parameter ITYPE, to convert the gray level values of the input image into data values. Then, it uses the equation

OUT_DATA = COEF(1) + COEF(2) * IN_DATA ** COEF(3)

to convert the input data values to output data values, where the COEFs are inputs from the user. Finally, equation (2) or (4), depending on the parameter OTYPE, is used to convert the output data values back to the gray levels.

PARAMETERS:
(1) **I_FIL** is the name of the input image file to be converted.
(2) **O_FIL** is the name of the output image file.
(3) **I_TYPE** specifies the image type of I_FIL. A "1" indicates a pigment image, a "2" indicates a linearly scaled image.
(4) **O_TYPE** specifies the image type of O_FIL. A "1" indicates a pigment image, a "2" indicates a linearly scaled image.
(5) **FACTOR** is the factor to divide the gray level values in I_FIL to get the data values. It is used only when it is greater than 0 and I_TYPE is 2. A value of 0 or less and I_TYPE is 2 will use the slope and intercept in the header of I_FIL for the conversion of gray level values to data values.

(6) **COEF** contains three coefficients to convert the input data values, not the gray level values, to the output data values. The conversion equation is

```
OUT_DATA = COEF(1) + COEF(2) * IN_DATA ** COEF(3)
```

(7) **RANGE** is the gray level ranges for the pixels in I_FIL to be converted for the output image file.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: L2MULT
DATE: 10/28/91
MENU: L2PROD

DESCRIPTION: L2MULT generates seven level-2 image files from the level-1 data. The level-2 products (# = 1 to 7, respectively) are subsurface upwelling water radiances at 440, 520 and 550 nm, aerosol radiance at 670 nm, pigment concentration, Rayleigh radiance at 440 nm and diffuse attenuation at 490 nm (Austin and Petzold, 1981). These files are labeled OUTFILE#.img where OUTFILE is the root name entered by the user and # is defined above. In addition, a text file with an OUTFILE root name and a "L2P" extension is created which has all the important constants and parameters used in processing the data. The program allows for options regarding the atmospheric correction algorithm, land, cloud and aerosol thresholds and water radiance scaling. The atmospheric correction algorithm is discussed in Gordon et al. (1988).

Due to the 640 KB memory limitation problem of DOS real mode, L2MULT was redeveloped under protected mode. L2MULT is a driver routine that accepts input parameters and writes to a temporary file and then invokes the protected mode program INTL2MLT.EXE which reads the input parameters from the temporary file and generates the level-2 products.

PARAMETERS:

(1) **INFILE** is the name of any one of the level-1 files. The program uses channels 1 through 5 in the processing so all data files must be located in the same disk directory and have the same filename convention as used by the programs TP2IMG and WINDOW.

(2) **OUTFILE** is the root name to use for the files generated. (See the main program help text for this program.)

(3) **ITERATE** determines which atmospheric correction algorithm is to be used. If 0, the standard Gordon et al. (1988) algorithm is applied. If 1, the iteration algorithm of Smith and Wilson (1981) is used. If the iteration algorithm is used, the user should try using Angstrom exponents equal to zero. The program assigns zero values to pixels which do not converge after 10 iterations. When ITERATE=1, an additional image of water radiance at 670 nm is created.

(4) **CORR** is the index of the correction method to use for calculating total radiances:
   1: Use factors and method (time and gain dependent) of R. Evans (Univ. of Miami).
   2: Use correction factors specified by FACTOR.

(5) **ILTOPT** specifies the ILT option: If "1", ephemeris data from the ILT record of the level-1 scene will be used. If "0", much of these data will be obtained from the documentation record or calculated by SEAPAK based on the location and time at the start of the scene.

(6) **PIGMENT** allows the user to specify one of two pigment algorithms. The first is the "branching" algorithm of Gordon et al. (1983):

1. if Lw(550)=<0, then P=46.34456 (saturated); else,
2. if Lw(443)>0.15,
   then P = A2 * (Lw(443)/Lw(550))**B2, (A)
   where log10(A2)=0.053 and B2=-1.71;
   if P>=1.5 and Lw(520)>0
   then P = A4 * (Lw(520)/Lw(550))**B4, (B)
   where log10(A4)=0.522 and B4=-2.44;
   if P<1.5, then use (A) above;
3. if Lw(443)=<0.15 and Lw(520)>0, then use (B) above;
4. if Lw(443)=<0.15 and Lw(520)=<0,
   then P=46.34456 (saturated);

where Lw represents the water-leaving radiance for the band of the specified wavelength (nm) and P is the pigment concentration in mg/m^3. The second choice is a three-channel algorithm provided by Dennis Clark (see Muller-Karger et al., 1990) which has the form,

P = 5.56 * [((Lw(443) + Lw(520))/Lw(550)]**(-2.252)

(7) **NORMWAT** determines if the water radiance values are to be normalized (Gordon and Clark, 1981) in which case specify "1". Enter "0" to specify output of subsurface water radiance images. The calculation of the two radiance values are as follows:

(1) Obtain the Lw surface water radiance first:
Lt = Lr + La + t_up * Lw
t_up = exp(-(tauR/2 + tauOz)/cos(theta)) * ta
where Lt: total radiance
Lr: Rayleigh radiance
La: aerosol radiance
TauR : Rayleigh optical thickness
TauOz: ozone optical thickness
theta: satellite zenith angle
t_up : diffuse transmission factor (sensor)
ta : aerosol transmittance (1.0)

(2) Calculate the subsurface water radiance Lw_ss:
Lw_ss = Wref * Wref * Lw / (1.0 - Rho)
where Wref : refractive index of water (1.34)
Rho : Fresnel reflectance

(3) Calculate the normalized water radiance [Lw]:
[Lw] = Lw/cos(theta0)/t_down
t_down = exp(- (TauR/2 + TauOz)/cos(theta0))
where theta0 : solar zenith angle
t_down : diffuse transmission factor (sun)
The [Lw] calculated should be nearly independent of the solar zenith angle. For pigment concentrations less than 0.25 mg/m3, the values for 520nm and 550nm should be about 0.30 and 0.50, respectively.

(8) **LANCLD** is the channel 5 threshold in gray level value used to identify land and clouds. All pixels with values exceeding this value are flagged and assigned a value of 0 if MASKLC="1"

and the CLOUD threshold is not exceeded (i.e., it is not a cloud pixel).

(9) **CLOUD** is the channel 1 threshold used to identify clouds. If a pixel's gray level exceeds this value, the pixel was also flagged by LANCLD and MASKLC="1", the pixel will be assigned a value of 255. The program THRES can be used to determine the best land and clouds thresholds for a particular scene.

(10) **HAZE** is the channel 4 threshold used to flag high aerosol radiance pixels. If a pixel's gray level exceeds this value and MASKLC="1", the pixel will be assigned a value of 0.

(11) **MASKLC** determines whether or not pixels flagged by LANCLD, CLOUD and HAZE are assigned values of 0 and 255. If "0" is selected, the output values for the flagged pixels in the water radiance images will be the Rayleigh corrected values and the pigment image will have the channel 1 Rayleigh corrected values. "1" assigns values of 0 and 255.

(12) **ANGEXP** are the Angstrom exponents for bands 1 to 4, respectively. For the Gordon algorithm, the fourth value is not used.

(13) **WATER** determines the range of valid water radiances required for the purposes of scaling. The lower and upper values of RANGE will be mapped to 0 and 255 gray levels and other radiance values scaled linearly. All radiances resulting in gray levels less than one will be set to one and those resulting in gray levels greater than 254 will be set to 254.

(14) **DIFFUSE** determines the scaling range for the diffuse attenuation at 490 nm (K490). The equation to calculate the K490 is

```
K490 = 0.0883 * [Lw(443)/Lw(550)] ** (-1.491) + 0.022
```

and the gray level for the output image is determined as follows:

```
1. if (Lw(443) .le. 0.0) .or. (Lw(550) .le. 0.0)
   then GRAY_LEVEL = 0.
2. if (K490 .ge. DIFFUSE(2)) .or. (K490 .le. DIFFUSE(1))
   then GRAY_LEVEL = 0.
3. if (K490 .lt. DIFFUSE(2)) .and. (K490 .gt. DIFFUSE(1))
   then GRAY_LEVEL = min(max(1, RTMP), 254),  where
        RTMP = (K490 - DIFFUSE(1)) * RDIF,  and
        RDIF = 255 / (DIFFUSE(2) - DIFFUSE(1))
   (all K490 values between DIFFUSE(1) and DIFFUSE(2) will
   be assigned gray level values between 1 and 254)
```

(15) **OZONE** are optical thicknesses (in meters) for bands 1 to 4. If the value "-999" is entered, the values used will be from the PC-TOMS database for the day of the input CZCS scene and for the point nearest to the image center. If the PC-TOMS data point is missing or an error occurs accessing the data, a message to that effect will be displayed on the terminal along with the default values. The actual values used will be listed in the L2P log file. If defaults are used, the values will be 0.00106, 0.0144, 0.0279, and 0.0125. These thicknesses are the products of the absorption coefficients (3.4E-6, 46E-6, 89E-6, and 40E-6) used at the Univ. of Miami and an average amount of 313 Dobson units of ozone.

(16) **FACTOR** are the correction factors to use for calculating total radiances of bands 1 to 4, respectively. These will be used only when CORR=2.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: LATLON
DATE: 10/28/91
MENU: GEOGRAPH

DESCRIPTION: This program allows the user to locate specific latitude/longitude locations on a PC-SEAPAK image. The program initially uses the image currently displayed. The user can obtain the latitude/longitude of the current cursor position or move the cursor to a specified latitude/longitude. Once the desired latitude/longitude is obtained, the user has the capability to mark that location on the overlay graphics frame buffer or the frame buffer where the image is displayed. Once the points are marked, a line can be drawn connecting them. When the line is drawn, the distance, in kilometers, and the direction between these points is provided. Options are provided to drop new images into the frame buffers, to change the frame buffers for display, to list the current images in the frame buffers, to change the graphics palette and to toggle the current graphics palette and the cursor on/off.

PARAMETERS:
None.

DYNAMIC PARAMETERS:

I. Used for dropping a new image.

(1) **IMGFILE** is the name of the disk file containing the image to drop. This must be a standard 512x512x8 bit image with a known number of header blocks.

(2) **FRMBUF** is the index number (1-3) of the frame buffer to receive the image.

(3) **HEADNO** specifies the number of 512-byte header blocks in the new image. This number of blocks will be skipped before reading the image data.

(4) **YNIMG** is a flag indicating whether to display the dropped image (1) or leave on the current frame buffer being displayed (0).

II. Used for moving cursor to new latitude and longitude

(1) **LAT** is the latitude where the cursor is to be moved. The value may be in decimal degrees, DMS format or radians (see parameter UNITS).

(2) **LON** is the longitude where the cursor is to be moved. The value may be in decimal degrees, DMS format or radians (see parameter UNITS)

(3) **UNITS** is the units of LAT and LON :

  1. Decimal degrees (initial default value).
  2. DMS format, sDDDMMMSSS.SS, where s is for the sign, DDD is for degrees, MMM is for minutes and SSS.SS is for seconds of an arc (for example, -75030000.00 DMS is equal to -75.5 degrees, 163006000 is equal to 163.1 degrees).
  3. Radians.

Note that modulo arithmetic is used for all three types of units. For example, -100.0, 260.0, 620.0, etc., are all equivalent degrees and may be entered for 100 west longitude.

FUNCTION KEY DEFINITIONS:

ESC: Exits the program.

F1: Marks the current cursor location on the overlay frame buffer with the color defined by the current graphics palette. Note, only 200 points can be marked for each graphics palette (1-7) selected.

F2: Erases the marks, defined by the current graphics palette, starting with the most recent. When the most recent mark has been erased, the next to the most recent will be erased when the button is depressed again, etc.

F3: Draws a vector between two previously marked points and determines the distance between them in kilometers and the angle from point 1 to point 2 measured from north.

F4: Erases the most recently drawn vector. After the most recent vector has been erased, depressing the button again erases the next most recent vector, etc.

F5: Changes the overlay graphics palette for marking the cursor location by increasing current palette number by one and if the value is greater than seven it will be reset to one.

F6: Displays the cursor location in pixel/line (TV coordinates) as well as in latitude and longitude.

F7: Marks the present cursor location on the current image frame buffer. The image data under the marked position are destroyed.

F8: Allows the user to specify a latitude and longitude to which to move the cursor on the image.

F9: Toggles the current graphics palette on/off.

F10: Toggles the cursor on/off.

ALT F1: Toggles the function key menu display on/off.

ALT F5: Lists the image file names loaded into frame buffers 1, 2 and 3.

ALT F6: Displays the next image frame buffer.

ALT F7: Allows the user to drop a new image into the frame buffer.

MOUSE LEFT BUTTON - Same as F1.

MOUSE RIGHT BUTTON - Same as ALT F1

PROGRAM NAME: LOGF
DATE: 10/28/91
MENU: SOFTFCT

DESCRIPTION: This program may be used to take the logarithm of an image file, pixel by pixel, according to the following general equation:

OUT = C * logB(I)

where OUT is the output data file designated by the parameter OFIL, C is a constant corresponding to the parameter CONST, logB is the logarithm to the base BASE, and I is the image data from the file IFIL. The image region of interest may be specified by GPAL and BFIL.

The calculation results are stored as real-valued data in OFIL in order to retain maximum accuracy. OFIL may be used subsequently as input to the program STATDIS in order to generate its image, optimize its gray scale, and save it as a disk image file. For a given pixel, if any I(n) value falls outside the RANGE values, or if an arithmetic error occurs during summation, OUT for that pixel will be flagged as "invalid" and subsequently assigned a value that is specified in STATDIS.

PARAMETERS:

(1) **IFIL** is the name of the input image file to be processed. The file should contain one header block (512 bytes) followed by 512 blocks of image data.

(2) **OFIL** is the name for the "data" file output to the disk. This file is composed of floating point numbers for higher accuracy. OFIL may be used as input to the program STATDIS in order to generate its image, optimize its gray scale, and save it as a PC-SEAPAK image file. Note, however, that the same blotch specification used in LOGF will be needed by STATDIS (i.e., the same blotch must be used unless GPAL=0). "Data" files such as OFIL cannot be dropped directly into the image display unit as images or used as input to this program. STATDIS must be used to generate and save image files from "data" files. In this way, you can interactively obtain, using STATDIS, an optimum gray scale for the image file corresponding to the range or subrange of data values in the "data" file. By convention, "data" file names end with the extension ".DAT" whereas image file names end with ".IMG". Note that the disk space required by a "data" file is proportional to the blotch area and may be much more than that required by an image file which is always 513 blocks. For a full image (GPAL=0, the equivalent of a full-image blotch), a "data" file will require 2049 blocks or about four times the space of an image file; for a blotch covering less than a quarter of the image, however, the "data" file will be smaller than an image file.

(3) **MODE** is a flag which indicates whether the pixel values of the IFIL image represents data (such as temperature or radiance) that are linearly related to gray levels, or pigment concentrations which are non-linear. A "1" should be entered for linear data and a "2" for pigment data.

(4) **RANGE** defines the range of pixel values in IFIL to be used for the computation in equation "OUT = C * logB(I)". The user should enter two values in the input data units. For a given pixel in IFIL, if its value falls outside the RANGE values, the corresponding pixel in OFIL will be flagged as "invalid." These "invalid" pixels may be assigned any value when using STATDIS to generate the image from OFIL. Again, the RANGE values must conform to the units of the IFIL image as specified by MODE and FACTOR (i.e. pigment concentration or units linearly proportional to gray levels). For example, to exclude only land and cloud pixels, the RANGE values should be 1.0 and 254.0 (the default values) for gray levels (MODE=1 and FACTOR=1) or 0.0409 and 44.46 for pigment concentrations (MODE=2).

(5) **BASE** is the logarithmic base to be used in equation "OUT = C * logB(I)" for processing each pixel. For a straight logarithm, set C=1 (the default value). BASE can be any number greater than one. Its default value is 10. Arithmetic errors may occur during calculations if inappropriate CONST, BASE, or input values are used. For example, errors will occur if OUT values are too large or too small or if zero or negative input image pixel values are present. Output data values of pixels for which arithmetic errors have occurred will be flagged as "invalid" and may be assigned any desired value when using the program STATDIS. (See the documentation for the program STATDIS dealing with the parameter INVAL for further information). Such pixels cannot be distinguished from flagged as "invalid" because of restrictions which are described later. LOGF will display the number of pixels with such errors, if any have occurred, at the end of its processing. With the use of an appropriate blotch or values for RANGE, the responsible pixels may be excluded from the calculations. However, these arithmetic errors may indicate that your values for CONST and other input parameters are incorrect and should be changed.

(6) **CONST** is a number by which the logarithm of each pixel will be multiplied as shown in equation "OUT = C * logB(I)".

(7) **FACTOR** is a linear scale factor used only if MODE=1, i.e. when a linear data-to-gray scale mapping function for the IFIL image is used. If greater than zero, it will represent the factor by which to divide the gray values of IFIL pixels in order to convert them into actual data values; if zero or less, the slope and intercept for this mapping function will be obtained from the file header of the IFIL. In order to retain the gray values, enter 1 (the default value); for sea surface temperature (SST), enter 8; for water radiance data, enter 85; for aerosol radiance data, enter 100.

(8) **GPAL** is the graphics palette which defines the blotch area(s) of interest and is in the range -7 to 7. If the number entered is positive, pixels within the blotch will be considered. If the number is negative, pixels outside the blotch will be considered. Only blotches defined by this graphics palette (the absolute value of GPAL) of the blotch file BFIL will be used. If "0" is entered, the entire image area (512 x 512) will be used and BFIL will be ignored.

(9) **BFIL** is the name of the blotch file which defines the image area(s) of interest unless GPAL= 0. Only blotches defined by the graphics palette corresponding to GPAL will be used. Blotches may be drawn and saved as files using the programs BLOTCH and BPSAV.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: LOOP
DATE: 10/28/91
MENU: FRMBUF

DESCRIPTION: The program LOOP displays a specified set of frame buffers in a changeable time interval, one after the other in the order in which they are input. The initial time interval of looping is about 0.4 second. The mouse left and right buttons and the upper and down arrows are used to increase and decrease the time interval of looping as well as the function key F1 is used to freeze the looping.

PARAMETERS:
(1) **Frame Buffers** gives the numbers of the frame buffers to loop through. Only values from 0 to 3 are valid input.

FUNCTION KEY DEFINITIONS:
ESC: Exits the program.
F1: Interrupts the looping and displays on the terminal the name of the image file displayed on the monitor. The user can press any key to resume the looping.
Up Arrow: Increases the looping time interval by about 0.1 seconds.
Down Arrow: Decreases the looping time interval by about 0.1 seconds.
Mouse Left Button: Increases the looping time interval by about 0.05 seconds.
Mouse Right Button: Decreases the looping time interval by about 0.05 second.

PROGRAM NAME: LUTCVT
DATE: 10/28/91
MENU: VAXTOPC

DESCRIPTION: The program LUTCVT is used to convert a look-up table file from the VAX SEAPAK format into the PC-SEAPAK format. It allows SEAPAK look-up table files created on the VAX with the programs TABSAV or PAINT to be used on the PC with the program TABLOAD.

PARAMETER:
(1) **VAX_LUT** is the input look-up table file name to be converted. This file should be created using the VAX SEAPAK program TABSAV and transferred to the PC.
(2) **PC_LUT** is the name of the output file for the converted look-up table to be used in PC-SEAPAK.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: LUTMOD
DATE: 10/28/91
MENU: LUTCOLOR

DESCRIPTION: This program allows you to change the brightness and contrast of the displayed image by modifying the look-up tables. Any interval of pixel values between 0 and 255 may be selected and assigned a different range of output values (gray levels). Through successive iterations of the program, a piecewise linear function can be generated in the LUT. The capabilities of this program make it useful for a variety of tasks other than normal image enhancement which can often be done as readily using STRETCH. One common use for this program is to force the clouds of a CZCS level-2 pigment scene to be white. Clouds will sometimes come out black depending on how the land and cloud mask flags were set in L2MULT. This program allows the user to set black (zero) values to 255.

PARAMETERS:
None.

DYNAMIC PARAMETERS:

I. Used for modifying the look-up tables

(1) **IN_GLV** defines the starting and ending input pixel values to be mapped to another range of gray level values defined by OUT_GLV.

(2) **OUT_GLV** defines the starting and ending output gray level values that the input pixel values IN_GLV are mapped to.

(3) **COL_LUT** indicates which look-up tables (red, green or blue) are to be modified. A value of "Y" in the specified color field will cause the look-up table of that color to be modified. For a gray shade image, all the three look-up tables should be used.

II. Used for feeding the look-up table mapped image into another frame buffer

(1) **FRMBUF** is the frame buffer to accept the image fed back from the displayed image.

(2) **RGB_LUT** defines which look-up table (red, green or blue) is to be used to generate the output image in the feedback process. The look-up table selected here will be assigned to all three look-up tables in the output image so that the output image will display gray shades.

FUNCTION KEY DEFINITIONS:

ESC: Exits the program.

F1: Allows the user to modify the look-up tables. The user needs to enter the values for parameters IN_GLV, OUT_GLV and COL_LUT. The brightness and contrast of the displayed image will be changed depending on the values entered.

F2: Lists the pixel/gray level map on the screen.

F3: Allows the user to feed the look-up table mapped image into another frame buffer.

F4: Generates a X-Y graph of the pixel/gray level map.
F5: Clears the graph generated using key F4.
F6: Toggles the displayed image on/off.
ALT F1: Toggles the function key menu display on/off.
MOUSE RIGHT BUTTON : Same as ALT F1.

PROGRAM NAME: MAPIMG
DATE: 10/28/91
MENU: PROJECTN

DESCRIPTION: This program will project an unmapped image into any of 20 specified map projections. A list of these projections is provided under the description for the parameter PROJCTN. Many parameters are common to all the projections, but some are projection dependent. The user is prompted for these latter parameters after the projection has been selected. Parameters are provided for controlling various aspects of the projections, e.g. the location of a landmark on the projected images can be specified via its latitude/longitude, or the horizontal and vertical scales can be controlled (magnification or reduction). The program will normally require several minutes to run.

The USGS General Cartographic Transformation Package (GCTP) is used for the map projection transformation equations. For additional information on projections, methodology and terminology, see Snyder (1982).

Due to the memory limitation problem under DOS real mode, MAPIMG was developed under protected mode which would not be constrained by the DOS 640 KB memory limitation. MAPIMG is just a driver routine which accepts the input parameters, writes them to a temporary file and then invokes the protected-mode program INTMPIMG.EXE to read the input parameters and generate the projected image.

PARAMETERS:

(1) **INFILE** is the name of disk file containing the image to be mapped. This must be a registered, unmapped PC-SEAPAK image.

(2) **OUTFILE** is the name of the disk file to create for the projected image.

(3) **PROJCTN** indicates the projection to use in mapping the images. A number in the range of 1 to 20, corresponding to any of the following projections, must be entered:

1 Universal Transverse Mercator (UTM)
2 State Plane Coordinates (SPC)
3 Albers Conical Equal-Area
4 Lambert Conformal Conic
5 Mercator
6 Polar Stereographic
7 Polyconic
8 Equidistant Conic
9 Transverse Mercator
10 Stereographic
11 Lambert Azimuthal Equal-Area
12 Azimuthal Equidistant
13 Gnomonic
14 Orthographic
15 General Vertical Near-Side Perspective
16 Sinusoidal

17 Equirectangular
18 Miller Cylindrical
19 Van der Grinten
20 Oblique Mercator

After the program starts, one will be prompted for additional information depending on the projection chosen.

(4) **LL_1** is the latitude/longitude (in decimal degrees) of point 1 which is to appear at a specified PIXEL/LINE location of the projected images. The points defined by LL_1 and LL_2 will appear DELTA_P (vertical or horizontal) pixels apart. Unlike LL_1, the point at LL_2 need not even be one which is expected to appear in the output images. These points must have some vertical and horizontal separation in the projected images.

LL_1, LL_2, PIXEL, LINE, and DELTA_P are used to determine the scale (i.e., meters per pixel/line) of the projected images as well as the area of the earth's surface to appear within the image boundaries. If the default value -999.0 is entered for any of these parameters or only one value is entered for LL_1 or LL_2, a default scale will be used such that the entire input images appear within the output image limits. In this case, the output images will be top- and left-adjusted and the entire pixel or line range will be used.

(5) **LL_2** is the latitude/longitude (in decimal degrees) of a second point. As was mentioned above, the points defined by LL_1 and LL_2 will be DELTA_P (vertical or horizontal) pixels apart and, unlike LL_1, the point at LL_2 need not be one which is expected to appear in the output images.

(6) **PIXEL** is the (horizontal) pixel position at which the point defined by LL_1 is to appear in the projected images. Note that, for the MVP-AT display, the left edge is considered as pixel 1.

(7) **LINE** is the line position at which you would like the point defined by LL_1 to appear in the projected images. Note that, for the MVP-AT display, the top edge of the display is considered as line 1.

(8) **DELTA_P** is the separation in pixels on the projected images of the points defined by LL_1 and LL_2. If DELTA_P is positive, it will be used as the horizontal separation of the two points; if negative, it will be used as the vertical separation. Note that the sign of DELTA_P does not otherwise imply anything about the relative positions of the two points.

Although LL_2 need not be in the output images, DELTA_P must still represent the horizontal or vertical separation in pixels. This therefore means that DELTA_P may be greater than the pixel width or height of the output images. For a given LL_1/LL_2, a larger DELTA_P will cause a magnification, i.e. increase the projection scale resulting in more pixels per earth meter.

(9) **ASPECT** represents the Y-over-X aspect ratio of the output images. It may be used to change the scale in one direction

relative to the other. An ASPECT>1 causes vertical stretching whereas an ASPECT<1(but greater than zero) causes horizontal stretching.

It should be noted that, if ASPECT is not equal to 1 (the default value), certain characteristics which are part of a projection's definition may be changed. The projection of the output images would then not be strictly that which you specify with PROJCTN. Therefore, an ASPECT value different from 1 should be used only for special purposes.

(10) **PFLAG** is a parameter indicating whether or not to display program progress. A "1" should be entered if one would like the program to display messages at certain intervals to indicate the progress of the processing.

DYNAMIC PARAMETERS:
After the previous parameters have been selected, the user will be prompted for more parameters depending upon the choice for a projection. The list of all the parameters with their definitions are given below. The user will not be prompted for all these parameters for any one projection.

(1) **ZONE** has two uses depending on the projection that calls for it. If PROJCTN=1, ZONE is the UTM zone. The default value is the zone for the default value of LONG. If one enters a different ZONE, the value entered will be used. If one enters a different LONG, the zone for that longitude will be used if ZONE was not changed. If PROJCTN=2, ZONE is the SPC zone.

(2) **LATI** is the value of the latitude (in decimal degrees) of any point in the image. The default value is the latitudinal midpoint of the unmapped image.

(3) **LONG** is the value of the longitude (in decimal degrees) of any point in the image. The default value is the longitudinal midpoint of the unmapped image. LONG may be used to determine the UTM zone of the projection (see ZONE).

(4) **SPHEROID** specifies the standard Earth spheroid to use. Clarke 1866 is the default. The following integers with their corresponding spheroid may be entered:

1:	Clarke 1866;	2:	Clarke 1880;
3:	Bessel;	4:	New International 1967;
5:	International 1909;	6:	WGS 72;
7:	Everest;	8:	WGS 66;
9:	GRS 1980;	10:	Airy;
11:	Modified Everest;	12:	Modified Airy;
13:	Walbeck;	14:	Southeast Asia;
15:	Australian National;	16:	Krassovsky;
17:	Hough;	18:	Mercury 1960;
19:	Modified Mercury 1968;	20:	Sphere of R=6370997m

To use another spheroid, enter SPHEROID=0 and the desired values for MAJOR and MINOR.

(5) **MAJOR** is the semi-major axis for the desired spheroid in meters. If one inputs SPHEROID=0, then a value for this parameter must be entered along with a value for MINOR (the

semi-minor axis). The default value is the semi-major axis of the Clarke 1866 spheroid (SPHEROID=1) in meters.

(6) **MINOR** is the semi-minor axis for the desired spheroid in meters when SPHEROID=0. A value for MAJOR (the semi-major axis) must also be entered for this case. If MINOR=0 or MINOR=MAJOR and SPHEROID=0, a sphere of radius MAJOR will be used.

The default value is the eccentricity squared of the Clarke 1866 spheroid (the default spheroid, SPHEROID=1). The eccentricity squared (e2) is determined from:

e2 = 2f - f**2,

where f (the flattening) = 1 - (semi-minor axis/semi-major axis).

(7) **RADIUS** is the Earth's radius in meters. The default value is 6,370,997 meters.

(8) **LAT_0** is specific to the projection being utilized, but in all cases is to be entered in decimal degrees.

If PROJCTN is 3, 4, 7, 8, 9, or 20, it represents the latitude of the projection origin. For PROJCTN=20, the default value is the latitudinal midpoint of INFILE; otherwise, it is the INFILE latitude closest to the equator.

If PROJCTN is 5, 6, or 17, LAT_0 is the standard parallel (latitude of true scale). The default is the latitudinal midpoint of INFILE.

If PROJCTN is 10, 11, 12, 13, 14, or 15, it is the latitude at the center of the projection. The default value in all these cases is the latitudinal midpoint of INFILE.

If PROJCTN is 1, 2, 16, 18, or 19, LAT_0 is not used.

(9) **LONG_0**, for all projections except PROJCTN=20, is the longitude of the central meridian. The value entered should be in decimal degrees. The default value is the longitudinal midpoint of INFILE.

For PROJCTN=20, this parameter is used only if FLAG=1. The longitude of the point on the center line where AZIMUTH is measured should be entered. Since the default is FLAG=0, the default LONG_0 is zero for this case.

(10) **LAT_1** is specific to the projection being utilized, but in all cases is to be entered in decimal degrees.

If PROJCTN is 3, 4, or 8, the latitude of the first standard parallel should be entered. For PROJCTN=8, a second standard parallel is required if FLAG=1 (default). The default value is the latitude at 1/6th of the input image from its northernmost point. If the equator is in the input image and the northernmost point is at a smaller absolute latitude than the southernmost point, the default is the latitude at 2/6th the input image from its northernmost point.

If PROJCTN=20, LAT_1 is only used if FLAG=0 (default). It then represents the latitude of the first point used to define the center line. The default value is the northernmost point of the input image.

LAT_1 is not used in the other projections.

(11) **LONG_1** is only used if FLAG=0 (default) and represents the longitude (in decimal degrees) of the first point used to define the center line. The default value is the longitudinal midpoint of the input image.

(12) **LAT_2** is specific to the projection being utilized, but in all cases is to be entered in decimal degrees.

If PROJCTN is 3, 4, or 8, one should enter the latitude of the second standard parallel. For PROJCTN=8, this second standard parallel is used only if FLAG=1 (default). The default value is the latitude at 1/6th of the input image from its southernmost point. If the equator is in the input image and the southernmost point is at a greater, or equal, absolute latitude than the northernmost point, the default is the latitude at 2/6th the input image from its southernmost point.

If PROJCTN=20, LAT_2 is only used if FLAG=0 (default) and represents the latitude of the second point used to define the center line. The default value is the southernmost point of the input image.

LAT_2 is not used in the other projections.

(13) **LONG_2** is only used if FLAG=0 (default) and represents the longitude (in decimal degrees) of the second point used to define the center line. The default value is the longitudinal midpoint of the input image.

(14) **SCALE** is the projected image's central scale factor. The default value is 0.9996.

(15) **HEIGHT** is the height of the perspective point above the Earth's surface in meters. This parameter is used only for the General Vertical Near-Side Projection (PROJCTN=15). The default value is 950,000 meters which is the nominal altitude of the Nimbus 7 satellite.

(16) **AZIMUTH** is the azimuth angle (east of north) for the center line. This parameter is used only if FLAG=1 (The default is FLAG=0).

(17) **FLAG** is a parameter which is used only if PROJCTN is 8 (Equidistant Conic) or 20 (Oblique Mercator).

If PROJCTN=8, one should enter a "0" if one standard parallel is being specified or a "1" (the default), if two are being specified. See LAT_1, LONG_1, LAT_2, and LONG_2 for more clarification. Standard parallels are true to scale and free of angular distortion.

If PROJCTN=20, one should enter a "0" (the default) if one is defining the center line (which determines the obliqueness of the Oblique Mercator projection) by using the coordinates of two points in the image (LAT_1/LONG_1 and LAT_2/LONG_2) or a "1" if you are using the azimuth angle (AZIMUTH) at a point (LONG_0) to define the center line.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: MEANF
DATE: 10/28/91
MENU: SOFTFCT

DESCRIPTION: This program may be used to average disk image files, pixel by pixel, according to the following general equation:

```
OUT  =  C  +  sum[ W(n)*I(n)**E(n) ] / N     for n = 1 to NUM
```

where OUT is the output data file designated by the parameter OFIL, C corresponds to the constant CONST, W are the weights WEIGHT, I are the image data from the files IN_FIL, E are the exponents EXPONENT, N is the number of terms summed and NUM is the number of IN_FIL files. The image region of interest may be specified by GPAL and BFIL.

The calculation results are stored as real-valued data in OFIL in order to retain maximum accuracy. OFIL may be used subsequently as input to the program STATDIS in order to generate its image, optimize its gray scale, and save it as a disk image file. For a given pixel, if any I(n) value falls outside the RANGE values, or if an arithmetic error occurs during summation, OUT for that pixel will be flagged as "invalid" and subsequently assigned a value that is specified in STATDIS.

PARAMETERS:

(1) **IN_FIL** is the array of the input image file names to be processed. Up to 12 files may be entered at once. All the files should contain one header block (512 bytes) and followed by 512 blocks of image data.

(2) **WEIGHT** is the array of the weighing factors for IN_FIL. A number must be entered for each IN_FIL. Each number will be used as a multiplicative factor for the pixel values of its corresponding image (raised to the EXPONENT power) during summation. To illustrate the use, consider the following examples:
   1) for simple averaging, set CONST=0, WEIGHT(n)=1, and EXPONENT(n)=1;
   2) to raise a single image to the 3rd power, set CONST=0, WEIGHT(1)=1, and EXPONENT(1)=3.

(3) **EXPONENT** is the array of the exponents for IN_FIL. A number must be entered for each IN_FIL. Each number will be used as the power by which to raise the pixel values of its corresponding image during summation. Note that EXPONENT not equal to one will affect the units of their respective terms. It is the user's responsibility to ensure that the final units of terms are consistent. Arithmetic errors may occur during summation if inappropriate EXPONENT values are used. For example, errors will occur if EXPONENT is too large or too small, or if negative EXPONENT is used with zero or negative input image pixel values. Output data values of pixels for which arithmetic errors have occurred will be flagged as

"invalid" and may be assigned any desired value when using the program STATDIS. (See the documentation for the program STATDIS dealing with the parameter INVAL for further information). Such pixels cannot be distinguished from those flagged as "invalid" because of range restrictions which are described later. MEANF will display the number of pixels with such errors, if any have occurred, at the end of its processing. With the use of an appropriate blotch or values for RANGE, these pixels may be excluded from the calculations. However, these arithmetic errors may indicate that your values for EXPONENT and other input parameters are incorrect and should be changed.

(4) **OFIL** is the name for the "data" file output to the disk. This file is composed of floating point numbers for higher accuracy. OFIL may be used as input to the program STATDIS in order to generate its image, optimize its gray scale, and save it as a PC-SEAPAK image file. Note, however, that the same blotch specification used in MEANF will be needed by STATDIS (i.e., the same blotch must be used unless GPAL=0). "Data" files such as OFIL cannot be dropped directly into the image display unit as images or used as input to this program. STATDIS must be used to generate and save image files from "data" files. In this way, you can interactively obtain, using STATDIS, an optimum gray scale for the image file corresponding to the range or subrange of data values in the "data" file. By convention, "data" file names end with the extension ".DAT" whereas image file names end with ".IMG". Note that the disk space required by a "data" file is proportional to the blotch area and may be much more than that required by an image file which is always 513 blocks. For a full image (GPAL=0, the equivalent of a full-image blotch), a "data" file will require 2049 blocks or about four times the space of an image file; for a blotch covering less than a quarter of the image, however, the "data" file will be smaller than an image file.

(5) **MODE** is a flag which indicates whether the pixel values of the IN_FIL image(s) represent data (such as temperature or radiance) that are linearly related to gray levels, or pigment concentrations which are non-linear. A "1" should be entered for linear data and a "2" for pigment data.

(6) **RANGE** defines the range of IN_FIL pixel values to use for the summation. The user should enter two values in the input data units. For a given pixel location, if a value for any IN_FIL falls outside the RANGE values, the corresponding pixel in OFIL will be flagged as "invalid." These "invalid" pixels may be assigned any value when using STATDIS to generate the image from OFIL. Again, the RANGE values must conform to the units of the IN_FIL image(s) as specified by MODE and FACTOR (i.e. pigment concentration or units linearly proportional to gray levels). For example, to exclude only land and cloud pixels, the RANGE values should be 1.0 and 254.0 (the default values)

for gray levels (MODE=1 and FACTOR=1) or 0.0409 and 44.46 for pigment concentrations (MODE=2).

(7) **CONST** is a constant (in output data units) which is to be added to the summation. The user should enter a real number whose units match those of the other terms.

(8) **FACTOR** is a linear scale factor used only if MODE=1, i.e. when a linear data-to-gray scale mapping function for the IN_FIL image(s) is used. If greater than zero, it will represent the factor by which to divide the gray values of IN_FIL pixels in order to convert them into actual data values; if zero or less, the slope and intercept for this mapping function will be obtained from each file header of the IN_FIL disk image files. In order to retain the gray values, enter 1 (the default value); for sea surface temperature (SST), enter 8; for water radiance data, enter 85; for aerosol radiance data, enter 100.

(9) **GPAL** is the graphics palette which defines the blotch area(s) of interest and is in the range -7 to 7. If the number entered is positive, pixels within the blotch will be considered. If the number is negative, pixels outside the blotch will be considered. Only blotches defined by this graphics palette (the absolute value of GPAL) of the blotch file BFIL will be used. If "0" is entered, the entire image area (512 x 512) will be used and BFIL will be ignored.

(10) **BFIL** is the name of the blotch file which defines the image area(s) of interest unless GPAL= 0. Only blotches defined by the graphics palette corresponding to GPAL will be used. Blotches may be drawn and saved as files using the programs BLOTCH and BPSAV.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: MEM
DATE: 10/28/91
MENU: STAT1

DESCRIPTION: This program enables the user to obtain a power spectrum utilizing the maximum entropy method (MEM) for any region of a displayed image. The region is defined by the user (within this program) on a graphics palette and may have the shape of a line, rectangle (with horizontal and vertical sides relative to the display), or parallelogram. After a power spectrum is calculated, it may be plotted or output to an ASCII file as a list of the X/Y (frequency/spectral density) values. The algorithm used for calculating the power spectrum is from Press et al. (1986). Further information on the algorithm may be obtained directly from this source.

PARAMETERS:

(1) **MODE** specifies whether the type of the input data. A "1" (the default value) should be entered if the pixel values of the displayed image represent data that are linearly related to gray levels (such as radiance or temperature). A "2" should be entered if the data represents pigment concentrations (mg/m3).

(2) **FACTOR** is the linear conversion factor when MODE=1 and will be ignored for pigment input data (MODE=2). If FACTOR is positive, it will represent the factor by which to divide the gray values of the image pixels in order to convert them into actual data values. If a zero or negative number is entered, the slope and intercept for this mapping function will be obtained from the file header of the corresponding disk image file. Examples of specific values this parameter may assume are: 1) "1" (the default value) in order to retain the gray values; 2) "8" in order to obtain sea surface temperature (SST); 3) "85" in order to obtain water radiance data; and 4) "100" in order to obtain aerosol radiance data.

DYNAMIC PARAMETERS:

I. Parameters for calculating maximum entropy:

(1) **DIR** indicates along which direction to calculate the power spectrum (this is for a box or parallelogram only). A "1" is entered to indicate that the power spectrum is to be along the horizontal direction of a box or in the direction of the first defined side (i.e., between the first and second corners) of a parallelogram. A "2" is entered to indicate that the power spectrum is to be along the vertical direction of a box or the second side of a parallelogram (between the second and third corners). This parameter refers to the most recently defined area. The initial default value is 1.

(2) **RANGE** specifies the range of pixel values over which the power spectrum is to be calculated, i.e. which pixels will be used. Two values conforming to the units of the image (i.e. pigment

concentrations, gray levels or other linear scales) need to be entered. Pixels with values within this range, inclusively, will be used. The values of pixels outside the range will be replaced with an average of neighboring pixel values for the purposes of the MEM calculations. The initial default values are set so as to exclude absolute black or white pixels.

(3) **DETREND** indicates whether or not to detrend the data linearly before MEM calculations. A "YES" or "NO" should be entered as input. For box or parallelogram areas, each row of pixel data as specified by DIR is detrended individually. A least-squares fit through all the data in a row (or along a line in the case of a line is defined) is used to obtain the straight trend line. The initial default value is "YES".

(4) **DIST** is the distance, in kilometers, between adjacent image pixels along the Earth's surface . This parameter is used to convert the power spectrum frequencies into units of cycles per unit distance (cycles/km). The initial default value is 1.

(5) **POLES** represents the order of the maximum entropy method (MEM) approximation equation and, as such, its value should depend on the spectral characteristics of the data. The number of poles selected for the MEM calculations should be entered and be in the range of 1 to the number of pixels (N) minus 1 along the length of the area being studied. With larger POLES values, the spectral resolution of the power spectrum improves but spurious peaks may result and the computation time increases. As a compromise, the default value of 2*N/ln(2*N), calculated by the program, may be used. The following is from Press et al. (1986):

> In practice, one usually wants to limit the [poles] to a few times the number of sharp spectral features. With this restricted number of poles, the method will smooth the spectrum somewhat, but this is often a desirable property. If the number of poles or the number of data points is too large, round-off error can be a problem. With "peaky" data (i.e. data with extremely sharp spectral features), the algorithm may suggest split peaks even at modest orders, and the peaks may shift with the phase of the sine wave. Also, with noisy input functions, if you choose too high an order, you will find spurious peaks galore! Some experts recommend the use of this algorithm in conjunction with more conservative methods, like periodograms, to help choose the correct model order, and to avoid getting too fooled by spurious spectral features.

(6) **MIN_F** is the minimum frequency (cycles per unit distance) for the power spectrum. MIN_F may range from 0 to 0.5/DIST (the Nyquist frequency) but must be less than MAX_F. Note that the actual first frequency on the spectrum plot will be the first multiple of DELTA_F (see below) greater than or equal to MIN_F. The initial default value is 0.

(7) **MAX_F** is the maximum frequency (cycles per unit distance) for the power spectrum. MAX_F may range from 0 to 0.5/DIST (the Nyquist frequency) but must be greater than MIN_F. Note that the actual last frequency on the spectrum plot will be the first multiple of DELTA_F smaller than or equal to MAX_F. The initial default value is 0.5/DIST.

(8) **DELTA_F** is the difference (in cycles per unit distance) between adjacent frequencies for the power spectrum. The default value of DELTA_F is either (0.5/999)/DIST or (MAX_F-MIN_F)/299, whichever is greater. This default value represents the smallest value which you may use for DELTA_F. A larger value of DELTA_F saves computation time and may be adequate. Note that it is POLES, not DELTA_F, that determines the spectral resolution of the spectrum. However, DELTA_F should be small enough so that all peaks of interest are identifiable on the spectrum. After calculating the power spectrum, the total power of the spectrum and the variance of the input data will be displayed on the terminal. The total power is twice the sum of the spectral densities at the frequencies of the spectrum. These frequencies are all the multiples of DELTA_F from 0 to 0.5/DIST. Therefore, a power value much smaller than the variance indicates that significant peaks fell between spectrum frequencies and that a smaller DELTA_F should be used if possible.

II. Parameters for plotting the spectrum.

(1) **PLT_PAL** is the graphics palette to be used for the power spectrum graph.

(2) **YSCALE** is a flag indicating the type of scaling for the Y axis, i.e. linear or logarithmic. A "1" should be entered, if the power spectrum (Y axis) is to be in a linear scale. A "2" should be entered for a logarithmic scale. The initial default value is 1.

(3) **XLABEL** is the label for the X axis of the power spectrum graph with maximum of up to 40 characters. Upper and lower case letters and other characters may be used. The initial default label is "Frequency (cycles/km)"; subsequently, the previously entered label is used as the default. Note that the initial default label assumes that DIST is in kilometers.

(4) **YLABEL** is the label for the Y axis of the power spectrum graph with maximum of up to 40 characters. Upper and lower case letters and other characters may be used. The initial default label is "Spectral Density"; subsequently, the previously entered label is used as the default.

(5) **TITLE** is the title for the power spectrum graph which consists of up to 40 characters and appears below the power spectrum graph. Upper and lower case letters and other characters may be used. The initial default title is "MEM POWER SPECTRUM"; subsequently, the previously entered title is used as the default.

III. Parameters for outputting the plot data to a file.

(1) **O_FIL** is the output file name which will contain the X values (frequencies) in the first column and the corresponding Y

values (spectral densities) in the second column. Both columns are written in a FORTRAN 1PG15.7 format.

FUNCTION KEY DEFINITIONS:

ESC: Exits the program MEM.

F1: Defines line segment(s) over which the power spectrum is to be obtained. More than one line segment may be used in order to approximate a curved line.

F2: Defines a box (a rectangle with horizontal and vertical sides, i.e. sides along the pixel or line direction) over which the power spectrum is to be obtained.

F3: Defines a parallelogram over which the power spectrum is to be obtained. Three corners will need to be specified in a clockwise or counter-clockwise direction using the mouse left and/or right buttons.

F4: Asks the user to enter parameters (see DYNAMIC PARAMETERS I) for, and starts calculations of, the maximum entropy.

F5: Asks the user to enter parameters (see DYNAMIC PARAMETERS II) for, and starts calculations of, the power spectrum plot.

F6: Outputs the plot data of a power spectrum to an ASCII file. The parameter O_FIL (see DYNAMIC PARAMETERS III) will be requested.

F7: Turns all the graphics palettes on/off.

F8: Turns the displayed image on/off.

F9: Increases the current graphics palette by 1 or resets it to 1 if the value is greater than 7. The current graphics palette is used for defining the line segments (F1), the box (F2) or the parallelogram (F3).

F10: Erases all graphics defined by the current graphics palette.

ALT F1: Toggles function key menu display on/off.

MOUSE RIGHT BUTTON: Toggles function key menu display on/off.

PROGRAM NAME: MERGE
DATE: 10/28/91
MENU: IMGFILE

DESCRIPTION: MERGE is a program that allows one to concatenate up to five contiguous CZCS scenes. These images must not be mapped or have different orbit numbers. Since the total number of lines for an image cannot exceed 512, the program allows the user to specify a line subsampling factor to keep the final merged image within this limit. A scenario for using this program would be the following:

1) Determine the start line number in the northernmost scene one is interested in,
2) determine the total number of lines one wants to merge,
3) determine the subsampling or reduction factor required to encompass the desired area within one 512 line image,
4) use MERGE to make the merged scene from the input images.

PARAMETERS:

(1) **INFIL** is a list of the input files to be merged. This can include up to five contiguous scenes. The first file must be the northern most scene, the second the next most northern, etc.

(2) **OUTFIL** is the name of the merged image.

(3) **LINE** is the line number in the first image at which the merging begins. This line number is given in terms of screen coordinates and becomes the first line of the merged image.

(4) **REDL** is the line sampling reduction factor, i.e. a value of 1 means to take every succeeding line, 2 means to take every other line, etc.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: MODHDR
DATE: 10/28/91
MENU: HEADER

DESCRIPTION: This program allows the user to update or add information to the header block of PC-SEAPAK image files. Such information includes the control point file name, a code for the source of the image data, the year and day of the image, the orbit number, the starting pixel and line, the ending pixel and line, the intercept and slope for the gray-to-data conversion, and the minimum and maximum latitude and longitude of the image. Note that this program should be used with EXTREME CAUTION since header changes may impact the results of other PC-SEAPAK programs that use this information.

PARAMETERS:

(1) **IFIL** is the file name that specifies an image or a group of images for which the header block(s) will be modified. The wild card characters "*" and "?" may be used to specify a group of files. "?" is used to replace a single character and "*" is used to replace multiple characters. The wild card should be used very carefully, since all the changes are based on the information from the first searched image file only, and those changes will affect all other files.

(2) **CTLFIL** is the control point file name which contains the navigation information for the image to be modified.

(3) **SOURCE** is the data source code for the image. Currently, valid codes include "C1" for CZCS level-1, "C2" for CZCS level-2, "A6" to "A9" for AVHRR NOAA-6 to NOAA-9", "AA" and "AB" for AVHRR NOAA-10 and NOAA-11, and "G" for gridded image.

(4) **SYEAR** is the starting year for a satellite image.

(5) **SDAY** is the starting Julian day for a satellite image.

(6) **ORBIT** is the orbit number for a satellite image.

(7) **SPIX** is the starting pixel of the image in a 512x512 display.

(8) **EPIX** is the ending pixel of the image in a 512x512 display.

(9) **SLIN** is the starting line of the image in a 512x512 display.

(10) **ELIN** is the ending line of the image in a 512x512 display.

(11) **SLOPE** is the slope for converting the image gray levels into geophysical values using the following scaling equation:

data = gray/SLOPE + INTCPT

(12) **INTCPT** is the intercept for converting the image gray levels into geophysical values using the following scaling equation:

data = gray/SLOPE + INTCPT

(13) **LATMIN** is the minimum latitude of the image.

(14) **LATMAX** is the maximum latitude of the image.

(15) **LONMIN** is the minimum longitude of the image.

(16) **LONMAX** is the maximum longitude of the image.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: MOSAIC
DATE: 10/28/91
MENU: MOSAIC

DESCRIPTION: The purpose of this program is to provide a way to generate a "mosaic" from input images where the portions of each image to be incorporated is determined by area of interest and gray level range. Each image is assigned a priority and a look-up table so that a low priority image can be overlaid with a higher priority image. An example would be to overlay a cloud image (e.g., CZCS band 5) in black and white onto a pigment image with a color look-up table. The program generates a three-band image from up to twelve images and one graphics file. To get the mosaic image, for each input image, there are the gray level mask range, the color look-up table file, and the mask file defined by the user. Any of the graphics palettes in the input graphics file may be turned off as well as be assigned to any color. The input images will be processed first in backward order and the graphics will be the last to process. The process steps make the graphics the highest priority to be outputted to the mosaic image, followed by the order of those input images. The output mosaic image contains three bands of image data. Each band has 512x512 bytes of data. The first band is the red component of the mosaic image, followed by the green band and the blue band. The mosaic image can be displayed as a true color image by using the program RGBDIS.

PARAMETERS:

(1) **IMG_FIL** contains the input image file names. Up to twelve file names may be entered. All the input files should be in standard PC-SEAPAK image format containing 512 bytes of header followed by 512x512 of image data. The input sequence determines the priority to the output mosaic image. The one entered first has the highest priority in generating the mosaic image. The one entered second has the second priority and so on. Note that the graphics in GRA_FIL, when specified, override all the input image data in generating the mosaic image.

(2) **IMG_MASK** specifies the lower and the upper bounds of the gray level masks for the corresponding IMG_FIL. For each input image, only those pixels with gray level within or beyond the mask ranges will be processed and outputted to the mosaic image. Each set of the lower and the upper bounds has to be both in positive values or negative values. The positive values specify the inclusive range and only pixels of the corresponding input image file with gray level values within that range will be considered for output. The negative values specify the exclusive range and only pixels of the corresponding input image file with gray level values fall outside the range will be considered for output. For example, if the input lower and upper bounds are 2 and 254, only pixels within 2 and 254 (inclusive) will be output to the mosaic. If the

input lower and upper bounds are -2 and -254, only pixels with gray level 0, 1 and 255 (excluding 2 and 254) will be output to the mosaic.

(3) **LUT_FIL** contains the look-up table (LUT) file names for each corresponding IMG_FIL. The LUT file includes the red, green and blue three look-up tables. Each look-up table has 256 entries and each entry has a value from 0 to 255. The 256 entries are used as the index for each image pixel to get its red, green and blue attributes from the look-up tables. The red, green and blue bands of the output mosaic image are generated from those red, green and blue attributes defined in LUT_FIL for each input image. The LUT_FIL files should be generated by the program TABSAV or PAINT. For the mosaic image to be displayed as a gray level in program RGBDIS, the LUT_FIL has to be blank.

(4) **BLO_FIL** contains the file names to be used for masking corresponding IMG_FIL. These files can be any image file, graphics file, or blotch file and should not contain the header block. All pixels in a BLO_FIL with gray level zero will mask (disable) the corresponding pixels in the corresponding IMG_FIL to be processed for output to the mosaic image. A blank BLO_FIL means there is no mask checking for the corresponding IMG_FIL. Note that for the input images, the mask checking is done before the range checking.

(5) **OUT_FIL** is the output mosaic image file name. This file contains three 512x512 bytes of red, green and blue bands data. The image can only be displayed as a true color image by using the program RGBDIS.

(6) **GRA_FIL** is the overlay graphics file name. The pixel data in this file will override those from the input images if there are overlaps for the output mosaic image. The graphics palettes used in this file may be turned off as well as be assigned to any color.

(7) **ON_OFF** contains flags for graphics in GRA_FIL defined by the seven graphics palettes to be outputted (a value of 1) or not to be outputted (a value of 0) to the mosaic image.

(8) **RED_PAL** contains the red attributes to be assigned to the seven graphics palettes. The color for each graphics palette is decided by the combination of attributes from corresponding RED_PAL, GREEN_PAL and BLUE_PAL. By default, red (255, 0, 0) green (0, 255, 0), yellow (255, 255, 0), blue (0, 0, 255), purple (255, 0, 255), aqua (0, 255, 255) and black (0, 0, 0) colors are used for graphics palettes 1 to 7.

(9) **GREEN_PAL** contains the green attributes to be assigned to the seven graphics palettes. The color for each graphics palette is decided by the combination of attributes from corresponding RED_PAL, GREEN_PAL and BLUE_PAL. By default, red (255, 0, 0) green (0, 255, 0), yellow (255, 255, 0), blue (0, 0, 255), purple (255, 0, 255), aqua (0, 255, 255) and black (0, 0, 0) colors are used for graphics palettes 1 to 7.

(10) **BLUE_PAL** contains the blue attributes to be assigned to the seven graphics palettes. The color for each graphics palette

is decided by the combination of attributes from corresponding RED_PAL, GREEN_PAL and BLUE_PAL. By default, red (255, 0, 0) green (0, 255, 0), yellow (255, 255, 0), blue (0, 0, 255), purple (255, 0, 255), aqua (0, 255, 255) and black (0, 0, 0) colors are used for graphics palettes 1 to 7.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: MULTF
DATE: 10/28/91
MENU: SOFTFCT

DESCRIPTION: This program may be used to multiply a constant with an disk image file or to do multiplication between two disk image files, pixel by pixel, according to the following general equation:

OUT = AC + (MC * I1**E1 * I2**E2)

where OUT is the output data designated by the file OFIL, AC is an additive constant and corresponds to the parameter ACONST, MC is a multiplicative constant corresponding to the parameter MCONST, I1 and I2 are the image data from the files IFIL1 and IFIL2, and E1 and E2 are the exponents represented by the parameters EXPON1 and EXPON2. The image region of interest may be specified by GPAL and BFIL.

The calculation results are stored as real-valued data in OFIL in order to retain maximum accuracy. OFIL may be used subsequently as input to the program STATDIS in order to generate its image, optimize its gray scale, and save it as a disk image file. For a given pixel, if the I1 or I2 value falls outside the RANGE1 or RANGE2 values, or if an arithmetic error occurs during calculation, OUT for that pixel will be flagged as "invalid" and subsequently assigned a value that is specified in STATDIS.

PARAMETERS:

(1) **IFIL1** is the name of the first input disk image file one wants to process, the I1 in equation

OUT = AC + (MC * I1**E1 * I2**E2)

The file should contain one header block (512 bytes) followed by 512 blocks of image data.

(2) **MODE1** is a flag which indicates whether the pixel values of the IFIL1 image represents data (such as temperature or radiance) that are linearly related to gray levels, or pigment concentrations which are non-linear. A "1" should be entered for linear data and a "2" for pigment data.

(3) **EXPON1** is the exponents for IFIL1 which is used as the power to raise for all the pixel values in IFIL1 prior to multiplication. Note that EXPON1 not equal to one will affect the units of IFIL1. It is the users responsibility to ensure that the final units of terms are consistent. Arithmetic errors may occur during calculation if inappropriate EXPON1 value is used. For example, errors will occur if EXPON1 is too large or too small. Output data values of pixels for which arithmetic errors have occurred will be flagged as "invalid" and may be assigned any desired value when using the program STATDIS. (See the documentation for the program STATDIS dealing with the parameter INVAL for further information).

Such pixels cannot be distinguished from those flagged as "invalid" because of range restrictions which are described later. MULTF will display the number of pixels with such errors, if any have occurred, at the end of its processing. With the use of an appropriate blotch or values for range, the responsible pixels may be excluded from the calculations. However, these arithmetic errors may indicate that your value for EXPON1 and other input parameters are incorrect and should be changed. For simple multiplication, set AC=0, MC=1, and E1=E2=1 in equation

OUT = AC + (MC * I1**E1 * I2**E2)

For division, a negative EXPON1 makes the IFIL1 a divisor.

(4) **FACTOR1** is the linear scale factor for IFIL1 used only if MODE1=1, i.e. when a linear data-to-gray scale mapping function for the IFIL1 is used. If greater than zero, it will represent the factor by which to divide the gray level values of IFIL1 pixels in order to convert them into actual data values; if zero or less, the slope and intercept for this mapping function will be obtained from the file header of the IFIL1 disk image files. In order to retain the gray values, enter 1 (the default value); for sea surface temperature (SST), enter 8; for water radiance data, enter 85; for aerosol radiance data, enter 100.

(5) **RANGE1** defines the range of IFIL1 pixel values to use for the multiplication. The user should enter two values in the input data units. For a given pixel in IFIL1, if its value falls outside the RANGE1 values, the corresponding pixel in OFIL will be flagged as "invalid." These "invalid" pixels may be assigned any value when using STATDIS to generate the image from OFIL. Again, the RANGE1 values must conform to the units of the IFIL1 image as specified by MODE1 and FACTOR1 (i.e. pigment concentration or units linearly proportional to gray levels). For example, to exclude the land and cloud pixels, the RANGE1 values should be 1.0 and 254.0 (the default values) for gray levels (MODE1=1 and FACTOR1=1) or 0.0409 and 44.46 for pigment concentrations (MODE1=2).

(6) **IFIL2** is the name of the second input disk image file one wants to process, the I2 in equation

OUT = AC + (MC * I1**E1 * I2**E2)

The file should contain one header block (512 bytes) followed by 512 blocks of image data. The user may multiply a image with a constant by leaving the IFIL2 blank.

(7) **MODE2** is defined as is MODE1 except it applies to IFIL2 and its corresponding parameters.

(8) **EXPON2** is defined as is EXPON1 except it applies to IFIL2 and its corresponding parameters.

(9) **FACTOR2** is defined as is FACTOR1 except it applies to IFIL2 and its corresponding parameters.

(10) **RANGE2** is defined as is RANGE1 except it applies to IFIL2 and its corresponding parameters.

(11) **OFIL** is the name for the "data" file output to the disk. This file is composed of floating point numbers for higher accuracy. OFIL may be used as input to the program STATDIS in order to generate its image, optimize its gray scale, and save it as a PC-SEAPAK image file. Note, however, that the same blotch specification used in MULTF will be needed by STATDIS (i.e., the same blotch must be used unless GPAL=0). "Data" files such as OFIL cannot be dropped directly into the image display unit as images or used as input to this program. STATDIS must be used to generate and save image files from "data" files. In this way, you can interactively obtain, using STATDIS, an optimum gray scale for the image file corresponding to the range or subrange of data values in the "data" file. By convention, "data" file names end with the extension ".DAT" whereas image file names end with ".IMG". Note that the disk space required by a "data" file is proportional to the blotch area and may be much more than that required by an image file which is always 513 blocks. For a full image (GPAL=0, the equivalent of a full-image blotch), a "data" file will require 2049 blocks or about four times the space of an image file; for a blotch covering less than a quarter of the image, however, the "data" file will be smaller than an image file.

(12) **ACONST** is a constant (in output data units) which is to be added to the multiplication term as shown by AC in equation

OUT = AC + (MC * I1**E1 * I2**E2)

(13) **MCONST** is a real number by which to multiply the input images. This is shown in equation

OUT = AC + (MC * I1**E1 * I2**E2)

where MC is the multiplicative constant MCONST.

(14) **GPAL** is the graphics palette which defines the blotch area(s) of interest and is in the range -7 to 7. If the number entered is positive, pixels within the blotch will be considered. If the number is negative, pixels outside the blotch will be considered. Only blotches defined by this graphics palette (the absolute value of GPAL) of the blotch file BFIL will be used. If "0" is entered, the entire image area (512 x 512) will be used and BFIL will be ignored.

(15) **BFIL** is the name of the blotch file which defines the image area(s) of interest unless GPAL= 0. Only blotches defined by the graphics palette corresponding to GPAL will be used. Blotches may be drawn and saved as files using the programs BLOTCH and BPSAV.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: MV
DATE: 10/28/91
MENU: STAT1

DESCRIPTION: This program enables the user to generate a mean image and a standard deviation image from up to 300 input image files. The mean image is generated by simply taking the average of all the input images on a pixel by pixel basis. Pixels with gray levels outside the specified range are considered invalid and are excluded from the averaging. In a similar manner, the standard deviation image is generated by calculating the standard deviations at each pixel for all the images. Again, only the valid pixels are utilized, i.e. pixels with gray levels inside the specified valid range. Note also that an output gray level of 0 is assigned to a pixel in the standard deviation image whenever it is a valid pixel and has a count of less than 2. Two additional images are also generated, one representing the total number of valid pixels and the other representing the ratio of the standard deviation to the mean.

PARAMETERS:

(1) **IMGFILS** are the input disk image file names. Up to 36 image file names may be specified. However, since the wild card (* or ?) file format is supported, up to 300 image files can be processed. All the mean and standard deviation calculations are based on these input files. Note that if there is only one file name entered for IMGFILS, the program will assume it is a text file and read as input the images listed in this file. All of the image files should have a header block.

(2) **F_MEAN** is the output mean image file name. The mean image is generated by simply taking the average of all the valid pixels at a particular pixel location for all the input images.

(3) **F_STD** is the output standard deviation image file name. This image is generated by simply taking the standard deviation of all the valid pixels at a particular pixel location for all the input images. Where the resulting count is less than 2, the standard deviation is assigned a value of 0. If F_STD is blank, a standard deviation image file is not created.

(4) **F_RATIO** is the output file name for the ratio (STD/MEAN) image. If F_RATIO is blank, a ratio image file is not created.

(5) **F_VALID** is the output image file name for the number of valid pixels processed. Each pixel in this output file represents the total number of valid data pixels used in the calculation of the mean at the corresponding pixel location. If F_VALID is blank, a valid-pixel image file is not created.

(6) **MODE** specifies whether to calculate the statistics for the output images based on gray level values or to convert first to pigment concentration values. One should enter a "1" for gray levels or a "2" for pigment concentrations.

(7) **MINRNG** is the minimum range for valid pixel identification. This value and MAXRNG should conform to the data units specified by the parameter MODE. Only pixels within the MINRNG and MAXRNG range are used in the statistical calculations. Land/clouds and extremely high/low data pixels can be easily excluded from the statistical calculations by entering proper values of MINRNG and MAXRNG.

(8) **MAXRNG** is the maximum range for valid pixel identification. This value and MINRNG should conform to the data units specified by the parameter MODE. Only pixels within the MINRNG and MAXRNG range are used in the statistical calculations. Land/clouds and extremely high/low data pixels can be easily excluded from the statistical calculations by entering proper values of MINRNG and MAXRNG.

(9) **INVALID** is the value to be assigned to those pixels for which none of the input values satisfied the MINRNG and MAXRNG criterion. This value should be in data units corresponding to the data type specified by MODE.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: NODSST
DATE: 10/28/91
MENU: MIAMI

DESCRIPTION: This program will generate an image file in PC-SEAPAK format from a multi-channel sea-surface temperature (MCSST) file distributed by the NASA Ocean Data System (NODS) at the Jet Propulsion Laboratory (JPL). The NODS input file (INFILE) is a weekly composite of sea-surface temperature (SST) data for the globe derived from the TIROS-N/NOAA satellite series' Advanced Very High Resolution Radiometer (AVHRR). INFILE represents an equirectangular image of the world (90 N to 90 S and 180 W to 180 E) having a width of 2048 and a height of 1024 grid points, each being about 20x20 km at the equator.

Each grid point contains the average of daytime MCSST values from NOAA National Environmental Satellite, Data, and Information Service (NESDIS) Global Retrieval Tapes for that area and that week. Ocean points with no satellite data have values that are interpolated using an interactive Laplacian relaxation technique. This file format and the methodology for generating it was developed at the Univ. of Miami's Rosensthiel School of Marine and Atmospheric Sciences.

PARAMETERS:

(1) **INFILE** is the name of the NODS SST disk file to be converted. (See the main help text for more information about such files.)

(2) **OUTFILE** is the name of the disk file to create for the PC-SEAPAK image generated from INFILE. The image created is equivalent to an unmapped PC-SEAPAK grid image. Such an image does not require a control point file and may be mapped using the program MAPIMG. INFILE is an equirectangular image of the world (90 N to 90 S and 180 W to 180 E) having a width of 2048 pixels and a height of 1024 lines. The OUTFILE image will also be equirectangular and its areal coverage will be determined by the CENTER and REDFAC input parameters.

(3) **CENTER** is the latitude and longitude (in degrees and in that order) of the center point for the region to extract. This point will be located at the center of the OUTFILE image generated. The areal coverage (boundaries) of this image will be determined by REDFAC.

INFILE is an equirectangular image of the world (90 N to 90 S and 180 W to 180 E) having a width of 2048 pixels and a height of 1024 lines. Therefore, to obtain an image of the world, enter CENTER=(0,0) and REDFAC=(4,4); the image will be 512 pixels by 256 lines. If CENTER=(0,0) and REDFAC=(4,2), the image generated would be 512 by 512, but the north-south direction will be stretched (greater resolution) relative the east-west direction in INFILE. As another example, to obtain an image at the full INFILE resolution centered on Bermuda, enter CENTER=(32,-66) and REDFAC=(1,1).

(4) **REDFAC** is the reduction factors for the pixel and line directions, in that order, of OUTFILE relative to the image represented by INFILE. Positive values indicate subsampling (reduction) whereas negative values indicates zooming-in (expansion) by pixel replication. Note that reduction in this sense indicates an increase in geographical coverage while expansion indicates a decrease. Values of -1, 0, or 1 all generate an OUTFILE image having a one-to-one correspondence of pixels with INFILE grid points. See the help text of CENTER for examples of using REDFAC.

(5) **DTYPE** is a flag that indicates the type of INFILE data to use for generating OUTFILE. INFILE grid points contain actual AVHRR derived SST data as well as interpolated SST values (see main help text). If DTYPE is 1, the actual and the interpolated values will be used. If DTYPE is 0, only actual values will be used and OUTFILE pixels corresponding to INFILE grid points with interpolated values will be set to black (0 gray level). (Land pixels are always set to black.)

The percent of actual data to actual and interpolated data for the area covered by the OUTFILE image will be displayed on the screen at the end of this program.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: OZONE
DATE: 10/28/91
MENU: ATMOS

DESCRIPTION: This program allows the user to retrieve the ozone data (in Dobson units) from the PC-TOMS data files (the PC version of Total Ozone Mapping Spectrometer VAX data files) and to compute the optical thickness for the CZCS bands of wavelength 443, 520, 550 and 670 nm. The PC-TOMS data files were generated from the VAX version by averaging every five points in the latitudinal direction, four points in the longitudinal direction, and every three daily values. The resulting data resolution is five degrees from latitudes 50S to 50N, ten degrees for latitudes greater than 50 and less than 70, and 20 degrees for latitudes greater than 70. Currently, the files include data for November 1978 to the end of 1986. The optical thickness is computed by the following equation:

Optical thickness(i) = alpha(i) * ozone in Dobson units

where the i represents the four CZCS bands of wavelength 443, 520, 550 and 670 nm and alpha values are from the University of Miami defined as follows:

alpha(1) = 3.4e-6, alpha(2) = 4.6e-5
alpha(3) = 8.9e-5, alpha(4) = 4.0e-5

PARAMETERS:
(1) **LAT** is an array of latitude positions. The corresponding array values of LAT, LON, YEAR, and DAY specify the locations and times of the ozone values. Note that, if the default value -9.999 is entered and any one of the corresponding LON, YEAR, or DAY values is not the default, the LAT value immediately preceding will be used.
(2) **LON** is an array of longitude positions. The corresponding array values of LAT, LON, YEAR, and DAY specify the locations and times of the ozone values. Note that, if the default value -9.999 is entered and any one of the corresponding LON, YEAR, or DAY values is not the default, the LAT value immediately preceding will be used.
(3) **YEAR** is an array of years. The corresponding array values of LAT, LON, YEAR, and DAY specify the locations and times of the ozone values. Note that, if the default value 0 is entered and any one of the corresponding LON, YEAR, or DAY values is not the default, the LAT value immediately preceding will be used. YEAR values may be entered as two or four digits (e.g., 79 or 1979).
(4) **DAY** is an array of Julian days. The corresponding array values of LAT, LON, YEAR, and DAY specify the locations and times of the ozone values. Note that, if the default value 0 is entered and any one of the corresponding LON, YEAR, or DAY

values is not the default, the LAT value immediately preceding will be used.

(5) **OUTFIL** is the file name for the output listing. A blank or "CON" displays the output on the screen. To send the output to the printer, the user may enter "LPT1" or "LPT2", depending on how the printer is connected to the system.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: PAINT
DATE: 10/28/91
MENU: LUTCOLOR

DESCRIPTION: The program PAINT pseudocolors the image in the specified frame buffer. This means that the look-up tables (LUT's) associated with the red, green and blue guns of the palette used by the specified frame buffer are modified in such a way as to give various colors for different input values. The number of colors, the input count range for each color block and the color of each block are selectable by the user. There is a default color setup for up to 32 color blocks which is defined in the file LKTBL.PAR under the SEAPAK directory. A histogram is also generated which shows the frequency of occurrence for each gray level value. This can sometimes be of help in selecting the breakpoints for the color blocks.

PARAMETERS:

(1) **FRMBUF** is the number of the frame buffer (0 to 3) to be pseudocolored.

(2) **LGREYLEV** is the minimum gray level to be pseudocolored. The range between LGREYLEV and HGREYLEV should always be greater than or equal to the number of color blocks BRK_NO and the valid input for them are between 0 and 255 inclusive.

(3) **HGREYLEV** is the maximum gray level to be pseudocolored. The range between LGREYLEV and HGREYLEV should always be greater than or equal to the number of color blocks BRK_NO and the valid input for them are between 0 and 255 inclusive.

(4) **BRK_NO** is the number of color blocks (different colors) in the color bar. This number must be a value in the range [1,32]. This parameter can be modified within the program by using function key F8.

DYNAMIC PARAMETERS:

I. Used for changing gray level ranges and number of color blocks.

(1) **LGREYLEV** see PARAMETERS section.

(2) **HGREYLEV** see PARAMETERS section.

(3) **BRK_NO** see PARAMETERS section.

II. Used for changing the breakpoints of color blocks.

(1) **GREY_LEVEL** is an array of breakpoints to be assigned in the color blocks. The values should be entered in order. The parameters LGREYLEV and HGREYLEV will be used to check the low range of the first color block and the high range of the last color block. For example, the LGREYLEV=20, HGREYLEV=200 and the array of GREY_LEVEL entered is (10, 40, 100, 150, 220) then only four color blocks (1 to 4) will be created with gray level ranges defined as (20-40), (41-100), (101-150), (151-200).

III. Used for saving color blocks and look-up tables.

(1) **LUTFIL2** is the output file name which the color block information and the look-up tables of red, green and blue defined by the color blocks will be saved to.

IV. Used for restoring color blocks and look-up tables

(1) **LUTFIL3** is the input look-up table file name to be loaded. Only the file created in program PAINT which contains both the color blocks information and the look-up tables can be used.

FUNCTION KEY DEFINITIONS:

ESC: Exits the program.

F1: Allows the user to modify the color of any color block. Once this key is pressed, the displayed image will disappear temporarily and only the color blocks and five text messages and six squares will display on the screen. The five messages tell the user how to change the color on the color block. In the six squares, the EXIT square is used to quit current mode, the OK square is used to assign the changed color to the color block, the BOX x square displays the changed color for color block x, the R, G, and B squares displays current intensities of the red, green and blue components of the color showed in the BOX x square. To change the color of a specified color block, first, move the cursor with the mouse to the specified color block and press the left button, then current color and its red, green and blue intensities as well as the color block number of the color block selected by the user will be displayed on the BOX x, R, G and B squares. At this time, the user can change the color on the BOX x square by moving the cursor to any of the R, G and B squares and, using the left and right mouse buttons or up and down keys, to increase or decrease the intensity of the red, green or blue component. After the color has been changed in the BOX x square, the user have to move the cursor to OK square and press the mouse left button to make the changed color to be saved in the color block. To quit the process of changing color on the color blocks, the user can press the Esc key or move cursor to the EXIT square and press the mouse left button.

F2: Rotates the colors of all defined blocks in a circular, shift left manner. The look-up tables are then modified accordingly.

F3: Toggles the color blocks on/off.

F4: Toggles between black/white and color in the image.

F5: Allows the user to pseudocolor only the image values corresponding to a selected color block. To examine a single color, one must first move the cursor to the desired color block and depress the key F5.

F6: One may save the present look-up tables (colors) on disk by depressing this key and entering a file name. The file will contain two bytes for the block number and 320 bytes for the intensities of red, green and blue components, the minimum and maximum gray level ranges of each of the color blocks up to a maximum of 32 color blocks. The file will also contain 768

bytes for the red, green and blue look-up tables, each having 256 entries, defined by the color blocks. This file can be reloaded using PAINT and TABLOAD.

F7: Colors saved previously may be reloaded by depressing this key and then entering the file name. Note, only files saved in program PAINT can be used in here, the files saved in program TABSAV cannot be used in here.

F8: Enables the user to modify the minimum and maximum gray level ranges to which the pseudocolor will be applied and the number of color blocks. The valid values for the minimum and maximum gray level ranges should be between 0 and 255 and the valid value for number of color blocks should be between 1 and 32. The new specified minimum and maximum ranges will then be evenly divided into the new specified total number of color blocks.

F9: One can change the breakpoints for the color blocks using this key. After depressing this key, the user is prompted with the parameter GREY_LEVEL. The user should enter the values desired for the gray level breaks. The endpoints are assumed to be 0 and 255 and are not to be entered (key F8 can redefine this range if need be) unless blocks having single values ranges of 0 or 255 are desired. Up to 32 points may be entered. The breakpoints defined here will overwrite the setup, both the color block numbers and the breakpoints, defined in key F8.

F10: Toggles the histogram on/off.

ALT F1: Toggles the function key menu display on/off.

MOUSE RIGHT BUTTON - Same as ALT F1.

PROGRAM NAME: PARL2MU
DATE: 10/28/91
MENU: L2PROD

DESCRIPTION: This program creates or modifies the parameter file for the program BHL2MULT (batch run mode of program L2MULT). The output file is a binary file with fixed record length. Each record in the file contains one input parameter data set to be used in generating the level-2 images. The maximum number of parameter data sets that can be generated is 10.

PARAMETERS:

I. Used for the first input screen.

(1) **PARFIL** is the parameter file name to be created or modified. If the file PARFIL already exists, this program allows the user to modify it, otherwise, it will create a new file for it. This file is a binary file and should be used only for program BHL2MULT.

II. Used for the second input screen.

(1) **INFILE** is the name of any one of the level 1 files. The program uses channels 1 through 5 in the processing so all data files must be located in the same disk directory and have the same filename convention as used by the programs TP2IMG and WINDOW.

(2) **OUTFILE** is the root name to use for the files generated. (See the main program help text for this program.)

(3) **ITERATE** determines which atmospheric correction algorithm is to be used. If 0, the standard Gordon et al. (1988) algorithm is applied. If 1, the iteration algorithm of Smith and Wilson (1981) is used. If the iteration algorithm is used, the user should try using Angstrom exponents equal to zero. The program assigns zero values to pixels which do not converge after 10 iterations. When ITERATE=1, an additional image of water radiance at 670 nm is created.

(4) **CORR** is the index of the correction method to use for calculating total radiances:
   1: Use factors and method (time and gain dependent) of R. Evans (Univ. of Miami).
   2: Use correction factors specified by FACTOR.

(5) **ILTOPT** specifies the ILT option: If "1", ephemeris data from the ILT record of the level 1 scene will be used. If "0", much of these data will be obtained from the documentation record or calculated by SEAPAK based on the location and time at the start of the scene.

(6) **PIGMENT** allows the user to specify one of two pigment algorithms. The first is the "branching" algorithm of Gordon et al. (1983):

```
1. if Lw(550)=<0, then P=46.34456 (saturated); else,
2. if Lw(443)>0.15,
       then  P = A2 * (Lw(443)/Lw(550))**B2,          (A)
       where log10(A2)=0.053 and B2=-1.71;
```

```
if P>=1.5 and Lw(520)>0
   then  P = A4 * (Lw(520)/Lw(550))**B4,       (B)
   where log10(A4)=0.522 and B4=-2.44;
   if P<1.5, then use (A) above;
3. if Lw(443)=<0.15 and Lw(520)>0, then use (B) above;
4. if Lw(443)=<0.15 and Lw(520)=<0,
   then P=46.34456 (saturated);
```

where Lw represents the water-leaving radiance for the band of the specified wavelength (nm) and P is the pigment concentration in mg/m^3. The second choice is a three-channel algorithm provided by Dennis Clark (see Muller-Karger et al., 1990) which has the form,

```
P = 5.56 * [((Lw(443) + Lw(520))/Lw(550)]**(-2.252)
```

(7) **NORMWAT** determines if the water radiance values are to be normalized (Gordon and Clark, 1981) in which case specify "1". Enter "0" to specify output of subsurface water radiance images. The calculation of the two radiance values are as follows:

```
(1) Obtain the Lw surface water radiance first:
    Lt = Lr + La + t_up * Lw
    t_up = exp(-(tauR/2 + tauOz)/cos(theta)) * ta
    where Lt: total radiance
          Lr: Rayleigh radiance
          La: aerosol radiance
          TauR : Rayleigh optical thickness
          TauOz: ozone optical thickness
          theta: satellite zenith angle
          t_up : diffuse transmission factor (sensor)
          ta   : aerosol transmittance (1.0)
(2) Calculate the subsurface water radiance Lw_ss:
    Lw_ss = Wref * Wref * Lw / (1.0 - Rho)
    where Wref : refractive index of water (1.34)
          Rho  : Fresnel reflectance
(3) Calculate the normalized water radiance [Lw]:
    [Lw] = Lw/cos(theta0)/t_down
    t_down =  exp(- (TauR/2 + TauOz)/cos(theta0))
    where theta0 : solar zenith angle
          t_down : diffuse transmission factor (sun)
```

The [Lw] calculated should be nearly independent of the solar zenith angle. For pigment concentrations less than 0.25 mg/m3, the values for 520nm and 550nm should be about 0.30 and 0.50, respectively.

(8) **LANCLD** is the channel 5 threshold in gray level value used to identify land and clouds. All pixels with values exceeding this value are flagged and assigned a value of 0 if MASKLC="1" and the CLOUD threshold is not exceeded (i.e., it is not a cloud pixel).

(9) **CLOUD** is the channel 1 threshold used to identify clouds. If a pixel's gray level exceeds this value, the pixel was also flagged by LANCLD and MASKLC="1", the pixel will be assigned a value of 255. The program THRES can be used to determine the best land and clouds thresholds for a particular scene.

(10) **HAZE** is the channel 4 threshold used to flag high aerosol radiance pixels. If a pixel's gray level exceeds this value and MASKLC="1", the pixel will be assigned a value of 0.

(11) **MASKLC** determines whether or not pixels flagged by LANCLD, CLOUD and HAZE are assigned values of 0 and 255. If "0" is selected, the output values for the flagged pixels in the water radiance images will be the Rayleigh corrected values and the pigment image will have the channel 1 Rayleigh corrected values. "1" assigns values of 0 and 255.

III. Used for the third input screen.

(1) **ANGEXP** are the Angstrom exponents for bands 1 to 4, respectively. For the Gordon algorithm, the fourth value is not used.

(2) **WATER** determines the range of valid water radiances required for the purposes of scaling. The lower and upper values of RANGE will be mapped to 0 and 255 gray levels and other radiance values scaled linearly. All radiances resulting in gray levels less than one will be set to one and those resulting in gray levels greater than 254 will be set to 254.

(3) **DIFFUSE** determines the scaling range for the diffuse attenuation at 490 nm (K490). The equation to calculate the K490 is

```
K490 = 0.0883 * [Lw(443)/Lw(550)] ** (-1.491) + 0.022
```

and the gray level for the output image is determined as follows:

```
1. if (Lw(443) .le. 0.0) .or. (Lw(550) .le. 0.0)
   then GRAY_LEVEL = 0.
2. if (K490 .ge. DIFFUSE(2)) .or. (K490 .le. DIFFUSE(1))
   then GRAY_LEVEL = 0.
3. if (K490 .lt. DIFFUSE(2)) .and. (K490 .gt. DIFFUSE(1))
   then GRAY_LEVEL = min(max(1, RTMP), 254),  where
        RTMP = (K490 - DIFFUSE(1)) * RDIF,  and
        RDIF = 255 / (DIFFUSE(2) - DIFFUSE(1))
   (all K490 values between DIFFUSE(1) and DIFFUSE(2) will
   be assigned gray level values between 1 and 254)
```

IV. Used for the fourth input screen.

(1) **OZONE** are optical thicknesses (in meters) for bands 1 to 4. If the value "-999" is entered, the values used will be from the PC-TOMS database for the day of the input CZCS scene and for the point nearest to the image center. If the PC-TOMS data point is missing or an error occurs accessing the data, a message to that effect will be displayed on the terminal along with the default values. The actual values used will be listed in the L2P log file. If defaults are used, the values will be 0.00106, 0.0144, 0.0279, and 0.0125. These thicknesses are the products of the absorption coefficients (3.4E-6, 46E-6, 89E-6, and 40E-6) used at the Univ. of Miami and an average amount of 313 Dobson units of ozone.

(2) **FACTOR** are the correction factors to use for calculating total radiances of bands 1 to 4, respectively. These will be used only when CORR=2.

V. Used for the fifth input screen.

(1) **SFLAG** is a flag to specify whether to continue or to stop adding or modifying more parameter data sets. A value of 1 will continue adding or modifying the parameter data set specified in IX_NO. A value of 0 will write the parameter data sets to the output file specified in PARFIL and stop this program.
(2) **IX_NO** is the index number of the parameter data set to be added or modified next time. The maximum number is 10.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: PARMPIMG
DATE: 10/28/91
MENU: PROJECTN

DESCRIPTION: This program creates or modifies the parameter file for the program BHMAPIMG (batch run mode of program MAPIMG). The output file is a binary file with fixed record length. Each record in the file contains one input parameter data set to be used for generating the remapped images. The maximum number of the parameter data sets that can be generated is 10.

PARAMETERS:

I. Used for the first input screen.

(1) **PARFIL** is the parameter file name to be created or modified. If the file PARFIL already exists, this program allows the user to modify it; otherwise, it will create a new file for it. This file is a binary file and should be used only for program BHMAPIMG.

(2) **OPTION** is a flag to allow the user to easily create or modify a parameter file with the same projection. If a "Y" is entered, the user will only be prompted for the input and output file names for each set. If a new parameter file is being created the prompt for projection parameters will appear for each input set. If an existing parameter file is being modified, the prompt for projection parameters will appear for the first set only, and the values entered will be used for subsequent sets. If a "N" is entered, the projection parameters will be prompted for each input set separately.

II. Used for the second input screen.

(1) **INFILE** is the name of disk file containing the image to be mapped. This must be a registered, unmapped PC-SEAPAK image.

(2) **OUTFILE** is the name of the disk file to create for the projected image.

(3) **PROJCTN** indicates the projection to use in mapping the images. A number in the range of 1 to 20, corresponding to any of the following projections, must be entered:

1 Universal Transverse Mercator (UTM)
2 State Plane Coordinates (SPC)
3 Albers Conical Equal-Area
4 Lambert Conformal Conic
5 Mercator
6 Polar Stereographic
7 Polyconic
8 Equidistant Conic
9 Transverse Mercator
10 Stereographic
11 Lambert Azimuthal Equal-Area
12 Azimuthal Equidistant
13 Gnomonic
14 Orthographic
15 General Vertical Near-Side Perspective

16 Sinusoidal
17 Equirectangular
18 Miller Cylindrical
19 Van der Grinten
20 Oblique Mercator

After the program starts, one will be prompted for additional information depending on the projection chosen.

(4) **LL_1** is the latitude/longitude (in decimal degrees) of point 1 which is to appear at a specified PIXEL/LINE location of the projected images. The points defined by LL_1 and LL_2 will appear DELTA_P (vertical or horizontal) pixels apart. Unlike LL_1, the point at LL_2 need not even be one which is expected to appear in the output images. These points must have some vertical and horizontal separation in the projected images.

LL_1, LL_2, PIXEL, LINE, and DELTA_P are used to determine the scale (i.e., meters per pixel/line) of the projected images as well as the area of the earth's surface to appear within the image boundaries. If the default value -999.0 is entered for any of these parameters or only one value is entered for LL_1 or LL_2, a default scale will be used such that the entire input images appear within the output image limits. In this case, the output images will be top- and left-adjusted and the entire pixel or line range will be used.

(5) **LL_2** is the latitude/longitude (in decimal degrees) of a second point. As was mentioned above, the points defined by LL_1 and LL_2 will be DELTA_P (vertical or horizontal) pixels apart and, unlike LL_1, the point at LL_2 need not be one which is expected to appear in the output images.

(6) **PIXEL** is the (horizontal) pixel position at which the point defined by LL_1 is to appear in the projected images. Note that, for the MVP-AT display, the left edge is considered as pixel 1.

(7) **LINE** is the line position at which you would like the point defined by LL_1 to appear in the projected images. Note that, for the MVP-AT display, the top edge of the display is considered as line 1.

(8) **DELTA_P** is the separation in pixels on the projected images of the points defined by LL_1 and LL_2. If DELTA_P is positive, it will be used as the horizontal separation of the two points; if negative, it will be used as the vertical separation. Note that the sign of DELTA_P does not otherwise imply anything about the relative positions of the two points.

Although LL_2 need not be in the output images, DELTA_P must still represent the horizontal or vertical separation in pixels. This therefore means that DELTA_P may be greater than the pixel width or height of the output images. For a given LL_1/LL_2, a larger DELTA_P will cause a magnification, i.e. increase the projection scale resulting in more pixels per earth meter.

(9) **ASPECT** represents the Y-over-X aspect ratio of the output images. It may be used to change the scale in one direction relative to the other. An ASPECT>1 causes vertical stretching whereas an ASPECT<1(but greater than zero) causes horizontal stretching.

It should be noted that, if ASPECT is not equal to 1 (the default value), certain characteristics which are part of a projection's definition may be changed. The projection of the output images would then not be strictly that which you specify with PROJCTN. Therefore, an ASPECT value different from 1 should be used only for special purposes.

(10) **PFLAG** is a parameter indicating whether or not to display program progress. A "1" should be entered if one would like the program to display messages at certain intervals to indicate the progress of the processing.

III. Used for the third input screen. Note, the parameters used for this screen will depend on the projection type selected.

(1) **ZONE** has two uses depending on the projection that calls for it. If PROJCTN=1, ZONE is the UTM zone. The default value is the zone for the default value of LONG. If one enters a different ZONE, the value entered will be used. If one enters a different LONG, the zone for that longitude will be used if ZONE was not changed. If PROJCTN=2, ZONE is the SPC zone.

(2) **LATI** is the value of the latitude (in decimal degrees) of any point in the image. The default value is the latitudinal midpoint of the unmapped image.

(3) **LONG** is the value of the longitude (in decimal degrees) of any point in the image. The default value is the longitudinal midpoint of the unmapped image. LONG may be used to determine the UTM zone of the projection (see ZONE).

(4) **SPHEROID** specifies the standard Earth spheroid to use. Clarke 1866 is the default. The following integers with their corresponding spheroid may be entered:

1:	Clarke 1866;	2:	Clarke 1880;
3:	Bessel;	4:	New International 1967;
5:	International 1909;	6:	WGS 72;
7:	Everest;	8:	WGS 66;
9:	GRS 1980;	10:	Airy;
11:	Modified Everest;	12:	Modified Airy;
13:	Walbeck;	14:	Southeast Asia;
15:	Australian National;	16:	Krassovsky;
17:	Hough;	18:	Mercury 1960;
19:	Modified Mercury 1968;	20:	Sphere of R=6370997m

To use another spheroid, enter SPHEROID=0 and the desired values for MAJOR and MINOR.

(5) **MAJOR** is the semi-major axis for the desired spheroid in meters. If one inputs SPHEROID=0, then a value for this parameter must be entered along with a value for MINOR (the semi-minor axis). The default value is the semi-major axis of the Clarke 1866 spheroid (SPHEROID=1) in meters.

(6) **MINOR** is the semi-minor axis for the desired spheroid in meters when SPHEROID=0. A value for MAJOR (the semi-major

axis) must also be entered for this case. If MINOR=0 or MINOR=MAJOR and SPHEROID=0, a sphere of radius MAJOR will be used.

The default value is the eccentricity squared of the Clarke 1866 spheroid (the default spheroid, SPHEROID=1). The eccentricity squared (e2) is determined from:

e2 = 2f - f**2,

where f (the flattening) = 1 - (semi-minor axis/semi-major axis).

(7) **RADIUS** is the Earth's radius in meters. The default value is 6,370,997 meters.

(8) **LAT_0** is specific to the projection being utilized, but in all cases is to be entered in decimal degrees.

If PROJCTN is 3, 4, 7, 8, 9, or 20, it represents the latitude of the projection origin. For PROJCTN=20, the default value is the latitudinal midpoint of INFILE; otherwise, it is the INFILE latitude closest to the equator.

If PROJCTN is 5, 6, or 17, LAT_0 is the standard parallel (latitude of true scale). The default is the latitudinal midpoint of INFILE.

If PROJCTN is 10, 11, 12, 13, 14, or 15, it is the latitude at the center of the projection. The default value in all these cases is the latitudinal midpoint of INFILE.

If PROJCTN is 1, 2, 16, 18, or 19, LAT_0 is not used.

(9) **LONG_0**, for all projections except PROJCTN=20, is the longitude of the central meridian. The value entered should be in decimal degrees. The default value is the longitudinal midpoint of INFILE.

For PROJCTN=20, this parameter is used only if FLAG=1. The longitude of the point on the center line where AZIMUTH is measured should be entered. Since the default is FLAG=0, the default LONG_0 is zero for this case.

(10) **LAT_1** is specific to the projection being utilized, but in all cases is to be entered in decimal degrees.

If PROJCTN is 3, 4, or 8, the latitude of the first standard parallel should be entered. For PROJCTN=8, a second standard parallel is required if FLAG=1 (default). The default value is the latitude at 1/6th of the input image from its northernmost point. If the equator is in the input image and the northernmost point is at a smaller absolute latitude than the southernmost point, the default is the latitude at 2/6th the input image from its northernmost point.

If PROJCTN=20, LAT_1 is only used if FLAG=0 (default). It then represents the latitude of the first point used to define the center line. The default value is the northernmost point of the input image.

LAT_1 is not used in the other projections.

(11) **LONG_1** is only used if FLAG=0 (default) and represents the longitude (in decimal degrees) of the first point used to define the center line. The default value is the longitudinal midpoint of the input image.

(12) **LAT_2** is specific to the projection being utilized, but in all cases is to be entered in decimal degrees.

If PROJCTN is 3, 4, or 8, one should enter the latitude of the second standard parallel. For PROJCTN=8, this second standard parallel is used only if FLAG=1 (default). The default value is the latitude at 1/6th of the input image from its southernmost point. If the equator is in the input image and the southernmost point is at a greater, or equal, absolute latitude than the northernmost point, the default is the latitude at 2/6th the input image from its southernmost point.

If PROJCTN=20, LAT_2 is only used if FLAG=0 (default) and represents the latitude of the second point used to define the center line. The default value is the southernmost point of the input image.

LAT_2 is not used in the other projections.

(13) **LONG_2** is only used if FLAG=0 (default) and represents the longitude (in decimal degrees) of the second point used to define the center line. The default value is the longitudinal midpoint of the input image.

(14) **SCALE** is the projected image's central scale factor. The default value is 0.9996.

(15) **HEIGHT** is the height of the perspective point above the Earth's surface in meters. This parameter is used only for the General Vertical Near-Side Projection (PROJCTN=15). The default value is 950,000 meters which is the nominal altitude of the Nimbus 7 satellite.

(16) **AZIMUTH** is the azimuth angle (east of north) for the center line. This parameter is used only if FLAG=1 (The default is FLAG=0).

(17) **FLAG** is a parameter which is used only if PROJCTN is 8 (Equidistant Conic) or 20 (Oblique Mercator).

If PROJCTN=8, one should enter a "0" if one standard parallel is being specified or a "1" (the default), if two are being specified. See LAT_1, LONG_1, LAT_2, and LONG_2 for more clarification. Standard parallels are true to scale and free of angular distortion.

If PROJCTN=20, one should enter a "0" (the default) if one is defining the center line (which determines the obliqueness of the Oblique Mercator projection) by using the coordinates of two points in the image (LAT_1/LONG_1 and LAT_2/LONG_2) or a "1" if you are using the azimuth angle (AZIMUTH) at a point (LONG_0) to define the center line.

IV. Used for the fourth input screen.

(1) **SFLAG** is a flag to specify whether to continue or to stop adding or modifying more parameter data set. A value of 1 will continue adding or modifying the parameter data set specified in IX_NO. A value of 0 will write the parameter data sets to the output file specified in PARFIL and stop this program.

(2) **IX_NO** is the index number of the parameter data set to be added or modified next time. The maximum number is 10.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: PJTCOL
DATE: 10/28/91
MENU: PAINTJET

DESCRIPTION: This program allows the user to select the 7 colors for the overlay and 16 colors for the displayed image to generate a hard copy of an image on the PaintJet printer using the program IMGPRINT. The seven overlay colors are used to replace the seven overlay graphics palettes if there are overlay graphics on the image. In IMGPRINT, the user has to break the gray level values of the image into sixteen or less intervals so that each interval can be assigned to a color selected here. There are 330 colors available for selection, displayed as color bars on the monitor with index numbers on the top and on the left. The colors selected here have to be saved into a file before they can be used in the program IMGPRINT to generate a hard copy on the PaintJet printer. The file PAINTJET.PAR in the SEAPAK directory contains the default colors for the overlay and the image and is used in this program initially to load the colors displayed on the monitor.

PARAMETERS:
None.

DYNAMIC PARAMETERS:
I. Used for changing the color indices for the PaintJet printer.
(1) **ROW_INDEX** is the row index array to be used with COL_INDEX to select the colors from the 330 sample colors for the overlay or the image. Valid values range from 1 to 33.
(2) **COL_INDEX** is the column index array to be used with ROW_INDEX to select the colors from the 330 sample colors for the overlay or the image. Valid values range from 1 to 10.
II. Used for loading the color indices from a file.
(1) **PJTFIL1** is the file name which contains the row and column indices of the seven overlay colors and the sixteen image colors to be loaded and displayed on the monitor. The file must have been created previously with the F4 function key in this program.
III. Used for saving the color indices into a file.
(1) **PJTFIL2** is the file name to save the seven overlay color indices and sixteen image color indices. This file is to be used by the program IMGPRINT to generate the hard copy of an image on the PaintJet printer.

FUNCTION KEY DEFINITIONS:
ESC: Exits the program.
F1: Changes any of the seven colors for the overlay.
F2: Changes any of the sixteen colors for the image.
F3: Loads the colors for the overlay and the image from a file. The file must have been created previously by using the F4 function key in this program.

F4: Saves the color indices for the overlay and the image into a file.

PROGRAM NAME: PLI
DATE: 10/28/91
MENU: LUTCOLOR

DESCRIPTION: The program PLI provides the capability to stretch the contrast of images on the MVP-AT display. The program allows the user to define any number of gray level ranges and apply a piecewise linear stretch to each range independently of other ranges. The mouse is used to control the stretching by modifying the range of the input and output gray levels. When the cursor is moved to the left or right, the intercept of the input-to-output gray-level relationship line is shifted left or right, respectively. When the cursor is moved up or down, the slope of the line increases or decreases, respectively.

PARAMETERS:
There are no parameters.

DYNAMIC PARAMETERS:
I. Parameters for changing the gray level ranges for stretching.
(1) **IN_RNG** is gray level range of the image to be stretched.
(2) **OUT_RNG** is the gray level range of the output image.
II. Parameter for saving the stretched image into a frame buffer.
(1) **FRMBUF** is the image frame buffer in which to save the stretched image.
III. Parameter for saving the stretched image into a file.
(1) **O_FIL** is the name of the file to create for saving the stretched image. A blank header block will be added at the beginning of the image file.

FUNCTION KEY DEFINITIONS:
ESC: Exits the program.
F1: Allows the user to change the input and output gray level ranges for stretching.
F2: Displays the next image frame buffer.
F3: Allows the user to specify a frame buffer in which to save the stretched image.
F4: Toggles the image data between positive (0 to 255) and negative (255 to 0) linear ramps. This key will reset the linear stretch to the original status.
F5: Allows the user to save the stretched image into a file.
F6: Toggles the pixel value/gray level graphics display on/off.
F7: Lists the look-up table used for current stretched image.
ALT F1: Toggles the function key menu display on/off.
MOUSE RIGHT BUTTON - Same as ALT F1

PROGRAM NAME: PSIMAGE
DATE: 10/28/91
MENU: HARDCOPY

DESCRIPTION: This program generates a postscript format file of an image to be printed on a postscript printer. The image to be printed can be stored in a file or in a frame buffer on the MVP-AT. An optional graphics overlay from another file or frame buffer can also be printed with the image. The file generated here may be sent to the postscript printer by using the DOS command "COPY FILENAME.EXT LPT1:". Note that since this program is developed for QMS ColorScript 100 model 10, the output file of a color image may not be printed on other postscript printers. However the output file of a gray shaded image should be printable by all postscript printers.

PARAMETERS:

(1) **IMAGE** is the source of the image to be printed. A blank value for this parameter means no image will be printed. If a value of "0", "1", "2" or "3" is entered, the image in the frame buffer of the number entered will be used. Otherwise it assumes an image from a file will be used.

(2) **OVERLAY** is the source of the overlay graphics to be printed with the image. A blank value for this parameter means no overlay will be printed. A value "0", "1" , "2" or "3" will use the frame buffer with the number entered for the overlay graphics. Otherwise, an overlay graphics from a file will be used.

(3) **PRT_FIL** is the output file specification. The output file created here can be sent to the postscript printer with the DOS command "COPY FILENAME.EXT LPT1:".

(4) **LUT_FIL** is used to determine the default gray level break point values for GRAY_LEVEL. If this parameter is blank and the image is from a file, 16 equally spaced gray level values from 0 to 255 will be used for the default 16 break points. If this parameter is blank and the image is from a frame buffer, the look-up table of that frame buffer will be used to determine the default break points. Otherwise, if this parameter is not blank, the program will use the look-up tables in the file to decide the default break points. The file entered here must be created using program TABSAV or PAINT.

(5) **S_LINE** is the starting line of the image to be printed. A value between 1 and 512 should be entered.

(6) **E_LINE** is the ending line of the image to be printed. A value between S_LINE and 512 should be entered.

(7) **H_OFFSET** is the horizontal paper offset in inches. Each row of the image will be shifted right H_OFFSET inches when printing.

(8) **V_OFFSET** is the vertical paper offset in inches. The paper will scroll up V_OFFSET inches before printing.

(9) **H_SIZE** is the horizontal paper size in inches for the output image.

(10) **V_SIZE** is the vertical paper size in inches for the output image.

(11) **IMG_HDR** is the number of header blocks in the image file. This is used only when a file name is entered for the parameter IMAGE.

(12) **OVI_HDR** is the number of header blocks in the overlay graphics file. This is used only when a file name is entered for the parameter OVERLAY.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: PSTIMG
DATE: 10/28/91
MENU: MIAMI

DESCRIPTION: This program generates an image file in PC-SEAPAK format from a "binned" ("postage-stamp" or PST) file of the University of Miami's DSP image analysis system. This file format is used in the global CZCS processing activity (Feldman et al., 1989). The image represented by the PST file is assumed to be an equirectangular image of the world (90 N to 90 S and 180 W to 180 E) having a width of 2048 pixels and a height of 1024 lines. Each of these pixels is referred to as a "bin" representing approximately 20x20 km at the equator. Each bin contains data from the corresponding pixels of the larger resolution image(s) used to generate the PST file.

PARAMETERS:

(1) **PSTIMG** is the input file name of the DSP PST image.

(2) **OUTIMG** is the output disk file name for the PC-SEAPAK image generated from the PST image. The image created is equivalent to an unmapped PC-SEAPAK grid image. Such an image does not require a control point file and may be mapped using the program MAPIMG. The PST image is assumed to be an equirectangular image of the world. The OUTIMG image will be based on that projection and its areal coverage will be determined by the CENTER and REDFAC input parameters.

(3) **BAND** is the band name to be used to extract the data from the PST image file. The band name is only used if OUT_TYPE is 1 or 2. Possible band names include "K490", "Chlor", "La670", "nLw550", "nLw520", and "nLw440". The band name prefixes "sum_" and "sum_squared_" are optional and upper or lower case letters may be used.

(4) **OUT_TYPE** is the index number for the type of the data to be contained in the OUTIMG image: 1, mean (default); 2, standard deviation; 3, pixels per bin; or 4, image count. The PST image is assumed to be an equirectangular image of the world (90 N to 90 S and 180 W to 180 E) having a width of 2048 pixels and a height of 1024 lines. Each of these pixels is referred to as a "bin" representing approximately 20x20 kilometers at the equator. Each bin contains data from the corresponding pixels of the larger resolution image(s) used to general image (index 3) will represent in gray levels the number of valid scene pixels used to obtain the data sums for each bin. A mean image (index 1) will be that of the band-name value divided by the pixels-per-bin for each bin. (The band name is specified by BAND.) Index 2 will generate an image of the standard deviation for each pixel in the corresponding mean image. An image count image (index 4) will generate an image whose gray values are proportional to the number of satellite scenes which contributed to the summation of the data for the corresponding bins.

(5) **CENTER** is the latitude and longitude (in degrees and in that order) of the center point for the region you would like to extract. This point will be located at the center of the OUTIMG image generated. The areal coverage (boundaries) of this image will be determined by REDFAC. The PST image is assumed to be an equirectangular image of the world (90 N to 90 S and 180 W to 180 E) having a width of 2048 pixels and a height of 1024 lines. Therefore, to obtain an image of the world, enter CENTER=(0,0) and REDFAC=(4,4); the image will be 512 pixels by 256 lines. If CENTER=(0,0) and REDFAC= (4,2), the image generated would be 512 by 512, but the north-south direction will be stretched (greater resolution) relative the east-west direction in DSP_IMG. As another example, to obtain an image at the full DSP_IMG resolution centered on Bermuda, enter CENTER=(32,-66) and REDFAC=(1,1).

(6) **REDFAC** is the pixel and line reduction factors for the OUTIMG relative to the PST image. Positive values indicate subsampling (reduction) whereas negative values indicates zooming-in (expansion) by pixel replication. Note that reduction in this sense indicates an increase in geographical coverage while expansion indicates a decrease. Values of -1, 0, or 1 all generate an OUTIMG image having a one-to-one correspondence of pixels with PSTIMG pixels (bins).

(7) **HEADER** is a flag to generate a header block in the OUTIMG. If "YES," a PC-SEAPAK image header (one record of 512 bytes) will be written at the beginning of OUTIMG. If "NO," OUTIMG will contain only the data of the specified image area. To generate a standard PC-SEAPAK image file, HEADER must be "YES" and OUT_SIZE must be (512,512). Otherwise, you will not be able to use OUTIMG as an image with other PC-SEAPAK programs.

(8) **SUBSAMPL** is used when REDFAC(1) or REDFAC(2) is greater than 1. In these cases, there may be more than one DSPIMG bin for each output pixel. If SUBSAMPL="YES", only the first such bin will be used for each output pixel regardless of whether or not it contains data. If SUBSAMPL="NO", the first bin containing data will be used in each case. Note that, although the logic is not strictly subsampling when SUBSAMPL= "NO", more of the input data will be used to generate the output image. For this case, the output image may appear slightly fuller than for SUBSAMPL="YES".

(9) **MIN_DATA** may be used to define the intercept of the equation used to linearly scale the PST file's data values to gray levels. Because MIN_DATA is used only for linearly-scaled data, it will be ignored for pigment input data when OUT_TYPE is 1 (mean) or 2 (standard deviation). When used, input data values less than MIN_DATA will be set equal to MIN_DATA. The scaling is done as follows:

```
      SLOPE = (MAX_DATA - MIN_DATA) / 255.0
 GRAY_LEVEL = nearest_integer[(DATA_VALUE - MIN_DATA) /
                                   SLOPE]
```

If the default value (-9999.) is entered for MIN_DATA and MAX_DATA, a set of "standard" slopes and intercepts are used according to the input band specified in the BAND and the value of OUT_TYPE:

1. If input band is for nLw440, nLw520, nLw550, or La670, and OUT_TYPE equals 1 or 2, SLOPE=0.01 and MIN_DATA=0.0.
2. If input band is for K490 and OUT_TYPE=1, SLOPE=0.001 and MIN_DATA=0.0.
3. If input band is for K490 and OUT_TYPE=2, SLOPE=0.01 and MIN_DATA=0.0.
4. If OUT_TYPE=3, SLOPE=5.0 and MIN_DATA=4.0.
5. If OUT_TYPE=4, SLOPE=1.0 and MIN_DATA=0.0.

If the default value is entered for MIN_DATA but not for MAX_DATA, the actual minimum of the extracted input data will be used for MIN_DATA and the entered value for MAX_DATA. If the default value is entered for MAX_DATA but not for MIN_DATA, the actual maximum will be used for MAX_DATA and the entered value for MIN_DATA. If the same numeric value is entered for each, the actual minimum and maximum values will be used to determine the slope and intercept. (Otherwise, an improper slope of zero would result.)

(10) **MAX_DATA** may be used to determine the slope of the equation used to linearly scale the PST file's data values to gray levels. Because MAX_DATA is used only for linearly-scaled data, it will be ignored for pigment input data when OUT_TYPE is 1 (mean) or 2 (standard deviation). When used, input data values greater than MAX_DATA will be set equal to MAX_DATA. The scaling is done as follows:

```
         SLOPE = (MAX_DATA - MIN_DATA) / 255.0
    GRAY_LEVEL = nearest_integer[(DATA_VALUE - MIN_DATA) /
                                  SLOPE]
```

If the default value (9999.) is entered for MIN_DATA and MAX_DATA, a set of "standard" slopes and intercepts are used according to the input band specified in the DSP_IMG name and the value of OUT_TYPE:

1. If input band is for nLw440, nLw520, nLw550, or La670, and OUT_TYPE equals 1 or 2, SLOPE=0.01 and MIN_DATA=0.0.
2. If input band is for K490 and OUT_TYPE=1, SLOPE=0.001 and MIN_DATA=0.0.
3. If input band is for K490 and OUT_TYPE=2, SLOPE=0.01 and MIN_DATA=0.0.
4. If OUT_TYPE=3, SLOPE=5.0 and MIN_DATA=4.0.
5. If OUT_TYPE=4, SLOPE=1.0 and MIN_DATA=0.0.

If the default value is entered for MIN_DATA but not for MAX_DATA, the actual minimum of the extracted input data will be used for MIN_DATA and the entered value for MAX_DATA. If

the default value is entered for MAX_DATA but not for MIN_DATA, the actual maximum will be used for MAX_DATA and the entered value for MIN_DATA. If the same numeric value is entered for each, the actual minimum and maximum values will be used to determine the slope and intercept. (Otherwise, an improper slope of zero would result.)

(11) **OUTSIZE** contains the pixel width and line length of OUTIMG. The default values are for a standard PC-SEAPAK image of 512x512. The width may range from 1 to 2048 and the length may range from 1 to 1024. To generate a standard PC-SEAPAK image file, HEADER must be "YES" and OUT_SIZE must be (512,512). Otherwise, you will not be able to use OUTIMG as an image with other PC-SEAPAK programs.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: PSTXRT
DATE: 10/28/91
MENU: MIAMI

DESCRIPTION: This program allows the user to extract a PC-SEAPAK image from a full resolution (2048 x 1024) PST file created by the program PSTIMG.

PARAMETERS:

(1) **I_FIL** is the input file name. This file must be a full resolution PST file created by the program PSTIMG.

(2) **O_FIL** is the file name for the output image.

(3) **WINDOW** defines, in conjunction with REDFAC, the area of the input image to use for generating a PC-SEAPAK image. WINDOW(1) and WINDOW(3) specify the positions of the first and last pixels, respectively, to use from each input image line and WINDOW(2) and WINDOW(4) specify the first and last line numbers, respectively.

(8) **REDFAC** are the pixel and line reduction factors. Positive values indicate reduction by subsampling whereas negative values indicate magnification by pixel replication. For example, an entry of (2,2) will create images half as wide in pixels and half as high in lines as the scene area defined by WINDOW; an entry of (-2,-2) will generate images twice as high and wide. Values of -1, 0, or 1 are equivalent and generate images having a one-to-one correspondence of pixels with the scene defined by WINDOW.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: READ
DATE: 10/28/91
MENU: DATA

DESCRIPTION: This program allows the user to retrieve data from the displayed image at the cursor position. The values may be read as gray levels or in geophysical units corresponding to the image data type specified by the parameter TYPE. The cursor shape may be changed into a cross hair to specify one pixel or into a square to specify an area from which to read values. Six different box sizes are provided from 3 by 3 to 13 by 13 pixels. The cursor box is moved by the mouse or cursor keys and can be placed anywhere on the image. Additional function keys are provided to: 1) output values to a disk file or printer, 2) display cursor location or move cursor to new lat/lon position, 3) mark the cursor with current overlay graphics palette, 4) change the overlay graphics palette, 5) list the images currently loaded in the frame buffers, 6) change the frame buffer being displayed, and 7) drop a new image.

PARAMETERS:
None.

DYNAMIC PARAMETERS:
I. Used for selecting image type (F5 is pressed):
(1) **TYPE** specifies the conversion method (counts to geophysical) used when obtaining image values. This index should correspond to the type of image being displayed. One of the following indices must be entered:
    0. Gray level values
    1. Pigment concentrations (mg/m3)
    2. AVHRR sea surface temperatures (deg C)
    3. CZCS sea surface temperatures (deg C)
    4. Total radiance values (mW/cm2-um-sr)
    5. Rayleigh radiance values (mW/cm2-um-sr)
    6. Water radiance values (mW/cm2-um-sr)
    7. Aerosol radiance values (mW/cm2-um-sr)
    8. Diffuse attenuation values
    9. Primary productivity values
    10. Epsilon values
    11. Bathymetry (m)
    12. STATDIS generated images
    13. User-specified, linearly-scaled values

If a "4" is entered, one will be a prompted for the correction option (COR_OPT) and the correction factor (FACTOR), if needed by the selected correction option. If a "13" is entered, one will be prompted for a slope (SLOPE) and intercept (INTCPT) to use when converting gray levels into data values.

II. Used for image data type 4 (Total Radiance):

(1) **COR_OPT** indicates the correction method (model) to use for converting gray levels into total radiances:

1. Evans calibration
2. User-specified correction factor (see FACTOR)

(2) **FACTOR** defines the radiance correction factor for the CZCS band (1 to 4) corresponding to the image. FACTOR is used only if COR_OPT=2; otherwise it is ignored.

III. Used for image data type 13 (user-specified, linearly-scaled)

(1) **SLOPE** is the slope value of the linear relationship which will be used to convert image gray levels into the data values:

DATA = (SLOPE * GRAY) + INTCPT

(2) **INTCPT** is the intercept value of the linear relationship which will be used to convert image gray levels into the data values. The equation is given above.

IV. Used for dropping a new image (F7 is pressed)

(1) **IMGFILE** is the name of the disk file containing the image to drop. This must be a standard 512x512x8 bit image with a known number of header blocks.

(2) **FRMBUF** is the index number (1-3) of the frame buffer to receive the image.

(3) **HEADNO** specifies the number of 512-byte header blocks in the image. This number of blocks will be skipped before reading the image data.

(4) **YNIMG** is a flag indicating whether to display the dropped image (1) or leave on the current frame buffer being displayed (0).

V. Used for setting output destination (F9 is pressed)

(1) **OUTPUT** selects where the output resulting from function F9 is to be directed. Enter a value of "0", "1" or "2" with the meanings as follow:

OUTPUT=0. The output will be displayed only on the terminal. If the output was previously going to the printer or a disk file, the printer or that file will be closed (stop writing output to printer or file).

OUTPUT=1. The output will be displayed at the terminal and sent to the printer port defined in PRT_PORT. If the output was previously going to a disk file, that file will be closed at this time.

OUTPUT=2. The output will be displayed on the terminal and written to a disk file which you will be able to display, edit, or print using DOS commands after exiting READ. When using this option, you must specify the file name in OUT_FILE. If the output was previously going to the printer, the printer will be closed at this time. If the output was going to a disk file whose name is different from OUT_FILE, that file will be closed. If it was going to a file of the same name, the output will

continue to be appended to that file. Note that once a file is closed, it cannot be reopened; if OUT_FILE is the same as that previously used for a file which has been closed or already existed on the disk, you will get an error message.

(2) **OUT_FILE** is the name of a disk file to receive the output. OUT_FILE is ignored if OUTPUT=0 or 1 but must be specified if OUTPUT=2. The output will continue to be written to OUT_FILE until the value of OUTPUT is changed or a new OUT_FILE is specified at which time OUT_FILE will be closed. Note that once a file is closed, it cannot be reopened; if OUT_FILE is the same as that previously used for a file which has been closed or already existed on the disk, an error message will be displayed.

(3) **PRT_PORT** is the printer port to receive the output. PRT_PORT is ignored if OUTPUT=0 or 2, but must be specified if OUTPUT=1.

VI. Used for moving cursor to new latitude and longitude

(1) **LAT** is the latitude where the cursor is to be moved. The value may be in decimal degrees, DMS format or radians (see parameter UNITS).

(2) **LON** is the longitude where the cursor is to be moved. The value may be in decimal degrees, DMS format or radians (see parameter UNITS)

(3) **UNITS** is the units of LAT and LON :

1. Decimal degrees (initial default value).
2. DMS format, sDDDMMMSSS.SS, where s is for the sign, DDD is for degrees, MMM is for minutes and SSS.SS is for seconds of an arc (for example, -75030000.00 DMS is equal to -75.5 degrees, 163006000 is equal to 163.1 degrees).
3. Radians.

Note that modulo arithmetic is used for all three types of units. For example, -100.0, 260.0, 620.0, etc., are all equivalent degrees and may be entered for 100 west longitude.

FUNCTION KEY DEFINITIONS:

ESC: Exits the program.

F1: Outputs the image values on the cursor position in terms of gray levels (0 to 255).

F2: Outputs the image values in geophysical units that are inside the cursor box. Other output includes standard deviations (with and without 0 and 255), and the screen coordinates of the box center.

F3: Toggles the cursor type between crosshair and box.

F4: Changes the size of the box cursor in a loop of sizes 3x3, 5x5, 7x7, 9x9, 11x11 and 13x13.

F5: Allows the user to change the image data type.

F6: Displays the cursor location in pixel/line (TV coordinate) as well as in latitude and longitude.

F7: Allows the user to drop a new image into the frame buffer.

F8: Displays the next image frame buffer.

F9: Changes the output destination. The user is prompted for where the output should go (OUTPUT) in addition to the terminal, i.e. to a file or to the printer. The disk file name (OUT_FILE) will also be requested (this is ignored if the output is not going to a file). The output will continue to be routed to this destination unless a change is requested.

F10: Changes the overlay graphics palette for marking the cursor location by increasing current palette number by one and if the value is greater than seven it will be reset to one.

ALT F1: Toggles the function key menu display on/off.

ALT F8: Allows the user to move the cursor to a new position defined by the latitude and longitude entered by the user.

ALT F9: Lists all the image file names loaded into the image frame buffers.

ALT F10: Marks the cursor at the current position using the color defined by the current overlay graphics palette.

MOUSE RIGHT BUTTON - Same as ALT F1.

PROGRAM NAME: <u>REGION</u>
DATE: <u>10/28/91</u>
MENU: <u>OVERLY</u>

DESCRIPTION: This program allows the user to define up to a hundred blotched regions over a displayed image based on the latitude/longitude coordinates. Each blotched region has to be defined by four corner coordinates entered as latitude/longitude pairs in either a clockwise or counter clockwise order. The blotched region is valid (and hence displayed) only if all four corners fall within the input image.

PARAMETERS:
(1) **FRMBUF** is the frame buffer of the displayed image which the regions will be generated on.
(2) **GPAL** is the graphics palette to plot the boundary of the region or to fill the region defined by PNT1, PNT2, PNT3 and PNT4.
(3) **PNT1** is the latitude and longitude of the first point to define a region. Note that PNT1, PNT2, PNT3 and PNT4 must be entered in either clockwise or counter clockwise order.
(4) **PNT2** is the latitude and longitude of the second point to define a region.
(5) **PNT3** is the latitude and longitude of the third point to define a region.
(6) **PNT4** is the latitude and longitude of the fourth point to define a region.

DYNAMIC PARAMETER:
(1) **RGN_IX** is the index number of a region to be modified, deleted or filled.

FUNCTION KEY DEFINITIONS:
ESC: Exits the program.
F1: Allows the user to define a new region. The boundary of this region will be plotted immediately if all four corner points lie inside the boundary of the image.
F2: Allows the user to modify a specified region. The RGN_IX will be prompted first and the GPAL, PNT1, PNT2, PNT3, and PNT4 of the specified region will be loaded allowing the user to modify. The boundary of the new region will be displayed right after the old region to be cleared on the display.
F3: Allows the user to delete a specified region. The RGN_IX will be prompted. Note that, on the display, all the overlapped areas of the specified region with other regions will also be cleared. The deleted region cannot be filled (F4) and listed (F6) but can still be modified (F2).
F4: Allows the user to fill in a specified region. Note that, on the display, all the overlapped areas of the specified region with other regions will be filled with the same color defined for the specified region.

F5: Fills in all the regions. On the display, if there are overlapped areas, the color will depend on the color defined by the last region which covers that area.

F6: Lists both the latitude/longitude and pixel/line coordinates of all regions.

F7: Toggles all the regions on the display on/off.

F8: Displays current cursor position.

ALT F1: Toggles the function key menu display on/off.

MOUSE RIGHT BUTTON - Toggles the function key menu display on/off.

PROGRAM NAME: REGIST
DATE: 10/28/91
MENU: GEOGRAPH

DESCRIPTION: REGIST is a program that allows the user to manipulate the position of as many as two images (main image and a window image). The program can be used to shift (X and Y) an image relative to another image or graphics overlay. The program is particularly useful for aligning an image with a coastline overlay and uses the mouse or cursor keys to interactively shift the image. To do this, the graphics overlay must be displayed prior to entering REGIST. In PC-SEAPAK, the frame buffer 0 is reserved for the overlay graphics and only frame buffer 1, 2 or 3 can be used for the main image. But, if the user wants to display the window image also, then only frame buffer 1 can be used for the main image and frame buffer 2 or 3 must be used for the window image. Note, the mouse is used to roam the main image (default), the window image, or both, and to change the window size. The function key F3 is used to set the function of the mouse movement.

PARAMETERS:
There are no parameters.

DYNAMIC PARAMETER:
I. Used for saving the registered image.
(1) **OUTIMG** is the output file name to save the registered image.
(2) **HDRFILE** is the image file name whose header will be copied to the output image file OUTIMG.
(3) **HDR_BLK** is the number of 512-byte blocks that make up the header to be copied from the file HDRFILE into OUTIMG. PC-SEAPAK images normally have a 1-block header.

FUNCTION KEY DEFINITIONS:
ESC: Exits the program.
F1: Displays the next image frame buffer. An error message will be issued if a window (second image) is displayed.
F2: Toggles the window image on/off. The window image can be displayed only when the main image is in frame buffer 1.
F3: Changes the function of the mouse. Selections are for roaming the main image, roaming the window image, changing the window size, or roaming the main image and the window image together.
F4: Allows the user to save the main image into a file.
F5: Switches the window image between upper left quadrant, upper right quadrant, lower left quadrant and lower right quadrant.
F6: Switches the frame buffer between 2 and 3 for the window image.
F7: Clears the left or right wrap around after the roaming of the main image. Note the wrap around part won't be saved into the file even it is not cleared when the key F4 is pressed.
F8: Displays the pixel movements of the main image in the X and Y directions.

F9: Allows the user to zoom the image displayed in a loop with magnification factors of 1, 2, 4, and 8.
ALT F1: Toggles function key menu display on/off.
MOUSE RIGHT BUTTON - Same as ALT F1.

PROGRAM NAME: RESCALE
DATE: 10/28/91
MENU: LUTCOLOR

DESCRIPTION: RESCALE is a program which applies a piecewise linear rescaling to an image from a frame buffer into another frame buffer. It allows the user to specify up to 16 ranges to be linearly rescaled. For each range, the user must enter the minimum and the maximum gray level values for the input frame buffer and the output frame buffer.

PARAMETERS:

I. Used for the first input screen.

(1) **INBUF** is the input frame buffer (0 to 3) to be rescaled.

(2) **OUTBUF** is the output frame buffer (0 to 3) to store the result.

II. Used for the second input screen.

(1) **IN_GRLV1** is the array of the minimum gray level values which is used with IN_GRLV2 to define the ranges in the input frame buffer to be rescaled.

(2) **IN_GRLV2** is the array of the maximum gray level values which is used with IN_GRLV1 to define the ranges in the input frame buffer to be rescaled.

(3) **OUT_GRLV1** is the array of the minimum gray level values which is used with OUT_GRLV2 to define the ranges to be rescaled from the ranges defined in IN_GRLV1 and IN_GRLV2.

(4) **OUT_GRLV2** is the array of the maximum gray level values which is used with OUT_GRLV1 to define the ranges to be rescaled from the ranges defined in IN_GRLV1 and IN_GRLV2.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: RGBDIS
DATE: 10/28/91
MENU: MOSAIC

DESCRIPTION: This program displays a true color image in red, green and blue bands. The three input bands may be stored in one file with data interlaced by bands or by pixels, or in three different files. The red, green and blue bands will be dropped into the MVP-AT frame buffers 0, 1, and 3 separately for a true color display. Optionally, an overlay graphics file may be loaded into frame buffer 2 and displayed on top of this true color image.

PARAMETERS:

(1) **RBAND** is the input file name containing the red, green and blue band data, or just the red band data, to be displayed for a true color image. If this file contains three bands of image data, the parameters GBAND and BBAND must be blank and FMT must be either 0 or 1. The file generated by the program MOSAIC may be used here with parameter FMT=0 (data interlaced by bands).

(2) **GBAND** is the input file name for the green band data. If a file name is entered here, the program assumes that the three input bands are from three different files and the parameter FMT is not used.

(3) **BBAND** is the input file name for the blue band data. If a file name is entered here, the program assumes that the three input bands are from three different files and the parameter FMT is not used.

(4) **FMT** is the data format selection used only when the input file (RBAND) contains the red, green and blue bands. A value "0" indicates that the three bands were stored interlaced by bands (i.e., starting with all the red band data, followed by all the green band data and all the blue band data). A value "1" indicates that the three bands were stored interlaced by pixels (i.e., red, green, blue for the first pixel followed by red, green, blue for the second pixel, etc).

(5) **SKIP** is the number of 512-byte blocks to skip in the input file(s) to display the true color image.

(6) **GFIL** is the file name of the overlay graphics to be displayed on top of the true color image.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: RING
DATE: 10/28/91
MENU: CZCSL2

DESCRIPTION: This program allows the user to mask out areas of sensor "ringing" on any CZCS level-2 product images. This ringing effect occurs when the sensor scans from bright areas, such as clouds, ice, or sand, onto darker (ocean) areas. Over the bright areas, the radiometer saturates and requires a certain amount of time to recover after scanning away from such areas. During this recovery period, abnormally high counts will be recorded, often in a periodic fashion (thus "ringing"). The CZCS level-1 band 4 and band 5 images are needed by the program to determine the saturated and the land or cloud pixels. The program checks each pixel from west to east (the scanning direction) to see if the pixel is a land or cloud pixel (a level-1 band 5 gray level greater than or equal to LANCLD) and if it is saturated (a level-1 band 4 gray level greater than or equal to SATGRAY). If so, and the gray-level difference between that pixel and the adjacent east pixel in the level-1 band 4 is greater than DELTA, then DISTANCE pixels after the test pixel will be masked out as being ringing affected. Note that all input images must be unmapped.

PARAMETERS:

(1) **SATFILE** is the CZCS unmapped level-1 band 4 (670 nm) image file which was used with the other level-1 bands to generate the level-2 image residing in FRMBUF. A pixel will be considered saturated if its gray level in this file is greater than or equal to SATGRAY. If a pixel is a land or cloud pixel (see LCFILE), is saturated, and the difference in gray levels between it and the adjacent east pixel in SATFILE is greater than or equal to DELTA, then DISTANCE pixels after this pixel will be masked out (assigned the value of MASKGRAY) as being ringing affected pixels in O_FBUF (interactive mode) or OUTFILE (command mode).

(2) **LCFILE** is the CZCS unmapped level-1 band 5 (750 nm) image file which was used with the other level-1 bands to generate the level-2 image residing in FRMBUF. A pixel will be considered a land or cloud pixel if its gray level in LCFILE is greater than or equal to LANCLD. (See SATFILE for more information.)

(3) **INFILE** is the CZCS unmapped level-2 image file to be processed for ringing masking. This parameter is required for the command mode with specified arguments in the command line. In the interactive mode, this parameter may be blank. In this case, the current displayed image on FRMBUF will be processed.

(4) **OUTFILE** is the output file name for the ringing masked image. This parameter is only required for the command mode with specified arguments in the command line. In the interactive mode, the user may use function key F2 to save the ringing masked image into a file.

(5) **SATGRAY** is the gray level for the saturated pixel test. A pixel will be considered as saturated if its gray level in SATFILE is greater than or equal to SATGRAY.

(6) **LANCLD** is the gray level for the land/cloud pixel test. A pixel will be considered as a land or cloud pixel if its gray level in LCFILE is greater than or equal to LANCLD.

(7) **DELTA** is used to test for the ringing effect. If a pixel is a land or cloud pixel (see LCFILE) and is saturated (see SATFILE), then if the difference in gray levels between it and the adjacent east pixel in SATFILE is greater than or equal to DELTA, DISTANCE pixels after this pixel will be masked out (assigned the value of MASKGRAY) as being ringing affected pixels in O_FBUF.

(8) **DISTANCE** is the number of pixels to be masked out in O_FBUF as being affected (unreliable) east of the pixel where the onset of a ringing effect is detected (see DELTA).

(9) **MASKGRAY** is the gray level to be assigned to the DISTANCE pixels in O_FBUF following the onset of each ringing effect.

(10) **FRMBUF** is the frame buffer in which the unmapped level-2 image to be processed for ringing resides. If the INFILE is specified, it will be loaded into this frame buffer first, otherwise, the current displayed image will be processed. Note that this parameter is required only for the interactive mode.

(11) **O_FBUF** is the frame buffer into which the FRMBUF image will be loaded after masking out ringing affected pixels. Note that this parameter is required only for the interactive mode.

DYNAMIC PARAMETERS:

I. Parameters for saving the ringing masked image.

(1) **OFIL** is the disk file name to create for the ringing masked image.

(2) **FRMBUF** is the frame buffer in which the ringing masked image resides. Normally this will be O_FBUF (the default value).

(3) **HDRFILE** is the image file name whose header will be copied to OFIL. By default, the file name in FRMBUF will be used. If this parameter is blank, or the specified file does not exist, or an error occurs while reading it, a blank header block will be used for OFIL.

II. Parameters for changing the input parameters.
Same as in PARAMETERS section.

FUNCTION KEY DEFINITIONS:

ESC: Exits the program.

F1: Performs the ringing masking. Each pixel is processed from left to right (west to east) on each line from top to bottom (norht to south). The resulting image will be loaded into O_FBUF with the ringing affected areas set equal to MASKGRAY. These areas will also be covered by the color defined for the graphics palette currently selected.

F2: Saves the ringing masked image into a file. The parameters of DYNAMIC PARAMETERS I will be requested.

F3: Allows the user to enter new parameters (see PARAMETERS) for ringing masking.

F4: Toggles the image display between the original level-2 image and the ringing masked image.

F5: Replaces the image line at which the cursor is located with the values of the line immediately above.

F6: Replaces the image line at which the cursor is located with the values of the line immediately below.

F7: Replaces the image line at which the cursor is located with the average of the values from the lines immediately above and below.

F8: Turns the blotches (ringing masks) defined by current graphics palette on and off.

F9: Increases the current graphics palette by 1 or resets it to 1 if the value is greater than 7. The current graphics palette is used to color over the ringing affected (masked) areas.

F10: Allows the user to use functions equivalent to those of the program READ.

ALT F1: Toggles the function key menu display on/off.

MOUSE RIGHT BUTTON - Same as ALT F1.

PROGRAM NAME: RLINE
DATE: 10/28/91
MENU: DATA

DESCRIPTION: RLINE allows one to read and plot the pixel values of an image along a line drawn as an overlay. The values may be read and plotted as gray levels or in geophysical units corresponding to the image data type specified by a parameter which can be changed during the RLINE session. Additional function keys are provided to: start new vector, define new vertex, delete vertex, connect vertices, erase connection, output values to a disk file or the printer, change the graphics palette, list the images currently loaded in the frame buffers, change frame buffer being displayed, drop a new image, display pixel/line and lat/lon positions, and move cursor to a new lat/lon position.

PARAMETERS:
There are no parameters.

DYNAMIC PARAMETERS:

I. Parameter for changing the image data type.

(1) **RTYPE** specifies the input image data type. An integer with a value of 0 through 4 should be entered. The values are:
  - 0 specifies the use of gray level values.
  - 1 specifies a CZCS chlorophyll channel,
  - 2 specifies a AVHRR sea surface temperature channel (in degrees Celsius) ,
  - 3 specifies a CZCS sea surface temperature channel (in degrees Celsius),
  - 4 specifies radiance values derived from the slope and intercept in the image header.

II. Parameters for sending output to a file or the printer.

(1) **OUTPUT** is the name of a disk file or LPT1 or LPT2 for printer to receive the output. If the disk file already exists, the user will be prompted to append the output to that file or to replace the old data in that file with the new data.

(2) **RTYPE** specifies the data type to be outputted. An integer with a value of 0 through 4 should be entered. The values are:
  - 0 specifies the use of gray level values.
  - 1 specifies a CZCS chlorophyll channel,
  - 2 specifies a AVHRR sea surface temperature channel (in degrees Celsius) ,
  - 3 specifies a CZCS sea surface temperature channel (in degrees Celsius),
  - 4 specifies radiance values derived from the slope and intercept in the image header.

(3) **RANGE** contains the minimum and the maximum values to be used for mean and variance calculation. The value entered should conform to the image data type specified in RTYPE.

III. Parameters for moving cursor to new latitude and longitude.

(1) **LAT** is the latitude where the cursor is to be moved. The value may be in decimal degrees, DMS format or radians (see parameter UNITS).

(2) **LON** is the longitude where the cursor is to be moved. The value may be in decimal degrees, DMS format or radians (see parameter UNITS)

(3) **UNITS** is the units of LAT and LON :

1. Decimal degrees (initial default value).
2. DMS format, sDDDMMMSSS.SS, where s is for the sign, DDD is for degrees, MMM is for minutes and SSS.SS is for seconds of an arc (for example, -75030000.00 DMS is equal to -75.5 degrees, 163006000 is equal to 163.1 degrees).
3. Radians.

Note that modulo arithmetic is used for all three types of units. For example, -100.0, 260.0, 620.0, etc., are all equivalent degrees and may be entered for 100 west longitude.

IV. Parameters for dropping a new image.

(1) **IMGFILE** is the name of the disk file containing the image to drop. This must be a standard 512x512x8 bit image with a known number of header blocks.

(2) **FRMBUF** is the index number (1-3) of the frame buffer to receive the image.

(3) **HEADNO** specifies the number of 512-byte header blocks in the new image. This number of blocks will be skipped before reading the image data.

(4) **YNIMG** is a flag indicating whether to display the dropped image (1) or leave on the current frame buffer being displayed (0).

V. Parameters for generating the plot.

(1) **MIN_RNG** specifies the minimum value to be plotted. The input should depend on the image data type.

(2) **MAX_RNG** specifies the maximum value to be plotted. The input should depend on the image data type.

(3) **XTYPE** specifies the unit of the x-axis. A value 1 will use the number of pixels along the line to be plotted, a value 2 will use the distance (in km) along the plotted line.

(4) **X_PAL** specifies the palette number to be used to define the color for the plot.

FUNCTION KEY DEFINITIONS:

ESC: Exits the program.

F1: Defines a new vertex for the line and marks it with the color defined by current graphics palette.

F2: Allows the user to erase vertices beginning with the most recent one. If the most recent one is erased, then the next most recent can be erased, etc. If there is a line connecting the erased vertex, it will also be erased.

F3: Finds two unconnected vertices starting from the last defined vertex and draw a line to connect them.

F4: Finds two connected vertices starting from the last defined vertex and remove the line connecting them.

F5: Starts new line segments that are not connected to the others. This allows the user to put several independent tracks on the display at the same time.

F6: Displays the cursor location in pixel/line (TV coordinate) as well as in latitude and longitude.

F7: This key allows the user to change the image data type.

F9: Displays the output on the screen. The output contains all the pixels' locations and the gray level values and the data values (if image data type is not 0) as well as the minimum, the maximum, the mean and the standard deviation values.

F10: This key changes the overlay graphics palette for marking the vertex and connecting the vertices by increasing current palette number by one and if the value is greater than seven it will be reset to one.

ALT F1: Toggles the function key menu display on/off.

ALT F5: This key enables the user to plot the data along the line defined by current vertices. The user will be prompted for the minimum and the maximum values to be plotted as well as the x-axis unit and the graphics palette to be used for the plot.

ALT F6: Allows the user to drop a new image into the frame buffer.

ALT F7: Displays the next image frame buffer.

ALT F8: This key allows the user to move the cursor to a new position defined by the latitude and longitude entered by the user.

ALT F9: Lists the image file names loaded into the image frame buffers.

ALT F10: Allows the user to sent the output to a file or to the printer.

MOUSE LEFT BUTTON - Same as F1.

MOUSE RIGHT BUTTON - Same as ALT F1.

PROGRAM NAME: SCATT
DATE: 10/28/91
MENU: STAT2

DESCRIPTION: This program generates a scatterplot for the corresponding areas of two images resident in the MVP-AT's frame buffers. The output may be directed to the overlay graphics or to a frame buffer. If a frame buffer is used, the intensity (gray level) of the scattergram's points will be proportional to the number of points with those values, i.e. the count. A scale factor may be used to control those intensities. The inside or outside of a blotch, or the full images, may be specified as the areas of interest. Either image may be of pigment concentrations or units linearly proportional to the gray scale. Ranges of values in corresponding units may be used to restrict the pixels to consider. Histograms may also be requested and will appear at the top and to the right of the scattergram. The program provides the option for labels on the axes as well as an annotation line below the scattergram. The scatterplot may also be overlaid with a grid if desired. Note that this program has to be run under DOS directly (as opposed to, e.g., under DESQview) so as to ensure that the histogram data are collected correctly.

PARAMETERS:

(1) **FBUF1** is the frame buffer for one of the two input images. The X axis of the scattergram will be used for the image in FBUF1 and the Y axis for the FBUF2 image. Note that, due to the limitation on MVP-AT hardware, FBUF1 must be 1.

(2) **FBUF2** is the frame buffer for the second image and will be used for the Y axis of the scattergram. Currently, only 2 and 3 are valid inputs.

(3) **MODE1** defines whether the image in FBUF1 is scaled linearly or is in pigment concentration. A value of "1" (the default value) should be entered if the pixel values of the FBUF1 image represent data (such as temperature) that are linearly related to gray levels. A value of "2" should be entered if they represent pigment concentrations (mg/m3).

(4) **MODE2** is similar to MODE1 but applies to the image in FBUF2.

(5) **FACTOR1** is a non-negative scaling factor which is used only if MODE1=1, i.e. the data-to-gray scale mapping function is linear for the FBUF1 image. It is ignored when MODE1=2. If FACTOR1 is positive, it represents the factor by which to divide the gray values of FBUF1 pixels in order to convert them into actual data values. If zero is entered, the slope and intercept for this mapping function will be obtained from the header of the disk file for the FBUF1 image. In order to retain the gray values, a "1" (the default value) should be entered ; for sea surface temperature (SST), enter 8; for water radiance data, enter 85; for aerosol radiance data, enter 100. Note that the use of different linear mapping functions does not alter the appearance of the scattergram or

histograms in any way other than ensuring that the values labelling the axes reflect those of the image data.

(6) **FACTOR2** is the linear, data-to-gray scale mapping function for FBUF2. Comments analogous to those of FACTOR1 apply here.

(7) **RANGE1** defines the range of FBUF1 pixel values to use for the determination of the scattergram and, if HIST=1 or 3, the FBUF1 histogram. Two values should be entered for this parameter. These values should conform to the units of the FBUF1 image (i.e. pigment concentration or units linearly proportional to gray levels) as specified by MODE1 and FACTOR1. Pixel values less than the smaller RANGE1 value and those greater than the larger RANGE1 value will be excluded from the plots. Therefore this range will determine the limits of the X axis and FBUF1 histogram. For example, to exclude land and cloud pixels for a level-2 CZCS image, the RANGE1 values should be 1.0 and 254.0 (the default values) for gray levels (MODE1=1 and FACTOR1=1) or 0.04093 and 45.0 for pigment concentrations (MODE1=2).

(8) **RANGE2** defines the range for valid FBUF2 values in data units. Comments analogous to those of RANGE1 apply here.

(9) **OBUF** is the frame buffer to use for the scatterplot and/or histogram output. If "0" (the default value) is entered, the frame buffer 0 will be used as an overlay graphics frame buffer for the output. Otherwise, the frame buffer entered here will be used as a regular image frame buffer for the output. In this case the intensity of the scattergram points may be used to indicate the pixel count for each point (see SCALE for more information). OBUF cannot be one of the input frame buffers FBUF1 and FBUF2.

(10) **BPAL** is the graphics palette containing the blotch area of interest. An integer in the range -7 to 7 should be entered. If the number is positive, only the pixels within the blotch will be considered; if the number is negative, only the pixels outside the blotch will be considered. If "0" is entered, the entire image area (512 x 512) will be used. This blotch, if used, must already be resident in the overlay graphics frame buffer since the option to create it within the program does not exist. You may use the program BPLOAD to load a previously generated blotch onto the overlay graphics frame buffer. Note that the blotch will be destroyed when running this program.

(11) **GPAL** is the graphics palette to be used to display the output when OBUF=0. Any value from 1 to 7 may be used. If OBUF is an image frame buffer, GPAL is ignored.

(12) **SCALE** is an intensity scale factor and is used when the output is directed to an image frame buffer (i.e., OBUF is not 0). For this case, the intensities of the plotted scattergram points will reflect the relative number of pixels (the count) represented by each point. Points with counts between the minimum and maximum values are given gray levels between 1 and 255 according to a linear scale where the mean is assigned to gray level 128 (half intensity). SCALE is used to multiply

each point's count before applying the scale to assign gray levels for plotting. If OBUF=0, SCALE is ignored. Note that, when HIST=3, the two histogram lines corresponding to each scattergram point may also be used as an indication of the count.

(13) **HIST** is a flag indicating whether or not histograms should also be generated. If HIST=0, no histogram is displayed (the scattergram is of course still generated). If HIST=1, only the FBUF1 histogram is displayed; if HIST=2, only the FBUF2 histogram is displayed; if HIST=3 (the default value), histograms of both FBUF1 and FBUF2 are generated along with the scattergram. The FBUF1 histogram is displayed above the scattergram (parallel to the X axis) and the FBUF2 histogram is displayed to the right of the scattergram (parallel to the Y axis). The values labelling each scattergram axis also apply to the corresponding histogram. The pixels used in the histogram generation are the same as those used in the scattergram and hence are also determined by RANGE1 and BPAL for FBUF1 (X axis) and by RANGE2 and BPAL for FBUF2 (Y axis). Note that each plotted scattergram point forms the intersection of two lines, one from each histogram, the sum of whose lengths indicate the relative number of pixels from the two images represented by that point.

(14) **XLABEL** is the text which will be used to label the X axis of the scatterplot. Up to 40 characters may be entered. The blank default value indicates that there will be no label for this axis.

(15) **YLABEL** is the text which will be used to label the Y axis of the scatterplot. As with the X axis, up to 40 characters may be entered, with the blank default value indicating that there will be no label for this axis.

(16) **ANNOT** is the annotation text that will be used for the caption on the bottom of the scatterplot. A maximum of 40 characters may be entered. The blank default value indicates that no annotation is to be displayed.

(17) **GRID** is a flag indicating whether or not the scatterplot should be overlaid with a grid. If GRID="YES" (the default value), the scatterplot will be displayed with a quartile grid. If GRID="NO", only the axes (drawn as a box) will be used.

FUNCTION KEY DEFINITIONS:

No function keys are used.

PROGRAM NAME: SCREEN
DATE: 10/28/91
MENU: ATMOS

DESCRIPTION: This program will display the image generated by the program CLRWAT whose gray levels represent the various criteria used in screening the image to identify "clear water" pixels. A color code is applied to clearly indicate the various types of pixels:

BLACK	:	Land, cloud or haze pixels.
PURPLE	:	Pixels at which the sun or scanner zenith angle is too high.
MAGENTA	:	Pixels whose aerosol radiances are outside valid range.
BLUE	:	Pixels for which the channel 2 (520 nm) and 3 (550 nm) normalized radiances are outside desired ranges.
YELLOW	:	Pixels whose pigment concentrations are too high.
GREEN	:	Pixels for which the epsilons are outside desired ranges.
ORANGE	:	Pixels for which the epsilons are non-monotonic.
WHITE	:	Clear-water pixels.

The sequence of the listed colors represents the order in which the criteria are applied and the pixels eliminated. For example, a magenta pixel would be designated as such if and only if the following results were obtained for these tests:

1. Land, cloud, or haze?	No.
2. Sun or scanner zenith angle too high?	No.
3. Aerosol pixel?	Yes.
4. No further test	

More detailed information about the actual criteria is given in the program CLRWAT.

PARAMETERS:
(1) **SCRFILE** is the input file name which was generated by CLRWAT and should have the extension "SCR".
(2) **FRMBUF** is the index number (1-3) of the frame buffer to receive the image.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: SELECT
DATE: 10/28/91
MENU: FRMBUF

DESCRIPTION: This simple program displays the selected frame buffer. If a graphics overlay is displayed, it will be applied to the selected frame buffer (even when the overlay frame buffer is selected).

PARAMETERS:
(1) **FRMBUF** is the index number (0 to 3) of the frame buffer to be displayed.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: SOLARZ
DATE: 10/28/91
MENU: AVHRRIN

DESCRIPTION: This program divides the geophysical values (linearly scaled to gray levels) of an image by the cosine of the solar zenith angle at each pixel. A new image file of the processed input file is generated. The primary application of this program is to correct percent albedo images of AVHRR visible channels 1 and 2 that are output by the programs TPAVHRR and DKAVHRR. This correction permits better comparisons of albedo images collected at different times of the day.

PARAMETERS:

(1) **INFILE** is the input image file name to be processed. The input file should be the albedo image of AVHRR visible channel 1 or 2 that is output by the program TPAVHRR or DKAVHRR.

(2) **OUTFILE** is the output file name for the processed input INFILE image.

(3) **GEO_MIN** is the minimum geophysical value (percent albedo) to use in conjunction with GEO_MAX for scaling each processed image's values to gray levels. If the flag value of -9999.0 (default) is entered, the actual minimum for the image will be used. A linear scale is used to map GEO_MIN and GEO_MAX into gray levels 1 to 254. Note that INFILE gray levels of 0 and 255 will be mapped to 0 and 255, respectively, for the OUTFILE pixels regardless of GEO_MIN and GEO_MAX. This is done so as not to convert invalid values clipped by these bounds into valid values in the output images.

(4) **GEO_MAX** is the maximum geophysical value (percent albedo) to use in conjunction with GEO_MIN for scaling each processed image's values to gray levels. If the flag value of -9999.0 (default) is entered, the actual maximum for the image will be used. A linear scale is used to map GEO_MIN and GEO_MAX into gray levels 1 to 254. Note that INFILE gray levels of 0 and 255 will be mapped to 0 and 255, respectively, for the OUTFILE pixels regardless of GEO_MIN and GEO_MAX. This is done so as not to convert invalid values clipped by these bounds into valid values in the output images.

(5) **WEITEK** specifies whether the Weitek floating point coprocessor is to be used for this program. A Weitek chip has to be installed inside the computer system for WEITEK="YES". WEITEK="NO", the default, indicates that the Intel 80387 coprocessor will be used to run this program.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: SPKSETUP
DATE: 10/28/91
MENU: INITIAL

DESCRIPTION: This program creates two files in the SEAPAK directory specified by parameter SPKPATH. The two files created are MVPAT.FIG which contains the information to configure the MVP-AT image board and SPKPATH.PAR which contains the path (directory) information for the CIA World Data Base files, the Navy's world bathymetry data files, NASA's Total Ozone Mapping Spectrometer (TOMS) data files, the HALO88 font files, and the HALO88 device driver for the MVP-AT. These two files are important to all the PC-SEAPAK programs. If they are missing or contain incorrect information, many of the PC-SEAPAK programs will not run correctly. The program INIT should always be run after any reconfiguration of the MVP-AT.

PARAMETERS:

(1) **SPKPATH** is the SEAPAK directory in which the two output files MVPAT.FIG and SPKPATH.PAR will be created. The initial directory for this parameter is derived from the environmental variable SEAPAK defined under DOS but is blank if such variable has not defined.

(2) **CIAPATH** is the directory where the CIA World Data Base files are stored. These files are used only by the program COAST. The initial directory for this parameter is retrieved from the file SPKPATH.PAR under the directory specified in SPKPATH but is blank if SPKPATH is blank or the file SPKPATH.PAR does not exist.

(3) **BATHPATH** is the directory where the Navy's world bathymetry data file is stored. Two versions of this file are available: a 10-minute resolution version normally distributed with PC-SEAPAK and a 5-minute resolution version available by request. This file is used only by the program BATHYIMG. The initial directory for this parameter is retrieved from the file SPKPATH.PAR under the directory specified in SPKPATH but is blank if SPKPATH is blank or the file SPKPATH.PAR does not exist.

(4) **TOMSPATH** is the directory where the NASA's Total Ozone Mapping Spectrometer (TOMS) data files are stored. The TOMS data files for PC-SEAPAK are a subset of the original TOMS data files used on VAX SEAPAK. These files are used by all the CZCS level-2 programs to calculate ozone optical thicknesses. The initial directory for this parameter is retrieved from the file SPKPATH.PAR under the directory specified in SPKPATH but is blank if SPKPATH is blank or the file SPKPATH.PAR does not exist.

(5) **HALOPATH** is the directory where HALO88's font files and device driver for MVT-AT are stored. HALO88's font files are used by such programs as ANNOTATE and GRID. The HALO88 library provides greater text handling capabilities than the MVP-AT

software library. The initial directory for this parameter is retrieved from the file SPKPATH.PAR in the directory specified in SPKPATH but is blank if SPKPATH is blank or the file SPKPATH.PAR does not exist.

(6) **MEMORY** is the memory address in hexadecimal of the MVP-AT image board. The initial value is retrieved from the file MVPAT.FIG in the directory specified in SPKPATH but is set to D000 (the default setup for the MVP-AT board) if it cannot be retrieved.

(7) **IO_ADDR** is the I/O address in hexadecimal of the MVP-AT image board. The initial value is retrieved from the file MVPAT.FIG in the directory specified in SPKPATH but is set to 300 (the default setup for the MVP-AT board) if it cannot be retrieved.

(8) **PIX_ASP** is used to determine the aspect ratio of the displayed pixels on the monitor. The initial value of this parameter is retrieved from the file MVPAT.FIG in the directory specified in SPKPATH but is set to 1 if it cannot be retrieved. A value of 1 will display the pixel with a 1:1 aspect ratio (square pixel) and a full-size image (512x512) will be a square area on the monitor. A value of 0 will display the pixel in 4:3 aspect ratio and a full-size image displayed on the monitor will have a rectangular shape. Note that this parameter is only valid for the interlaced signal from the MVP-AT which means that the parameter SCAN must be 1.

(9) **SCAN** is used to specify the scanning method of the MVP-AT. The initial value of this parameter is retrieved from the file MVPAT.FIG in the directory specified in SPKPATH but is set to 1 if it cannot be retrieved. A value of 1 will configure the MVP-AT to send an interlaced signal with a horizontal frequency of 15.75 KHz. A value of 0 will configure the MVP-AT to send a non-interlaced signal with a horizontal frequency of 31 KHz. This selection depends on the range of the horizontal frequency of the monitor used for the image display. If the available frequency range of the monitor covers both 15.75 KHz and 31 KHz, either the interlaced or non-interlaced scanning method may be selected.

(10) **MODE** is used to determine how many lines can be displayed from the MVP-AT. The initial value of this parameter is retrieved from the file MVPAT.FIG in the directory specified in SPKPATH but is set to 1 if it cannot be retrieved. A value of 1 will configure the MVP-AT to send a vertical frequency of 60 Hz and only 480 lines will be displayed on the monitor, assuming that the monitor also has the 60 Hz (American standard) vertical frequency. A value of 0 will configure the MVP-AT to send a vertical frequency of 50 Hz and 512 lines will be displayed on the monitor, assuming the monitor also has the 50 Hz (European standard) vertical frequency. Note that if the vertical frequency on the monitor does not match to the vertical frequency from the MVP-AT, the displayed image will jiggle. Most of the standard monitors sold in America have a vertical frequency range of 50 to 70 Hz with the default set to 60 Hz. Such monitors should be adjusted to 50 Hz so that 512-line

images can be displayed. This may be accomplished by adjusting the vertical-hold until a steady and clear image is obtained.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: STATDIS
DATE: 10/28/91
MENU: SOFTFCT

DESCRIPTION: This program allows the user to generate images from real-numbered PC-SEAPAK data files (e.g., those output by the programs ADDF, MEANF, DERIV, LOGF, etc.) and display them on specified MVP-AT frame buffers. The input files must all represent full images or have all been produced using the same blotch created by the graphics palette GPAL in the file BFIL. A function key menu will then be available to allow the user to save these input "data" files as image files. The available function keys vary depending on whether the input data files are linear (MODE=1) or pigment (MODE=2). For linear files, the function keys allow the user to obtain an optimum gray scale, i.e. a scale which shows the most structure for the areas or features of interest, to examine the minimum/maximum values, to generate image histograms so that one may choose more appropriate min/max values with which to rescale each image. Note that the pigment images (MODE=2) may not be rescaled.

PARAMETERS:
(1) **IFIL** are the names of real-numbered data files from which to generate images. All IFIL must have been generated by a PC-SEAPAK program (e.g., ADDF, MEANF, DERIV, etc.) using the same blotch area(s) defined in the blotch file BFIL by graphics palette GPAL, or, if GPAL=0, for a full image. An image corresponding to each file will be generated and displayed in the frame buffer that is entered in the corresponding parameter FRMBUF.
(2) **FRMBUF** are the frame buffers on which the corresponding IFILs will be generated and displayed. Thus, the image of the first IFIL file specified will be displayed in the first frame buffer specified, etc.
(3) **MODE** indicates whether the corresponding IFILs are linear or pigment files. MODE=1 indicates that a linear scale is used to generate gray level values for the image from the data values. The image may be rescaled by modifying the minimum or the maximum value of the data using the function key F4. MODE=2 indicates that a pigment scale is used to generate the image. When a pigment image is displayed, the function keys related to the rescaling of an image are removed from the function key menu and are disabled since such an image may not be rescaled.
(4) **INVAL** is the value in data units to be assigned to those pixels that were flagged as "invalid" in the programs which generated the input data files. This would normally have occurred when pixel values input to that program were outside of a specified range. STATDIS will then assign such pixels the corresponding INVAL value before converting them to gray levels in each image. By using a very small number (such as

-1.0E38) you can ensure that the invalid pixels are black (gray level 0) in the image regardless of the units of IFIL data. Conversely, you may enter a very large number (such as 1.0E38) in order to make such pixels white (gray level 255).

(5) **GPAL** is the graphics palette which defines the blotch of the area(s) of interest and is in the range -7 to 7. If the number entered is positive, pixels within the blotch will be considered. If the number is negative, pixels outside the blotch will be considered. Only blotches defined by this graphics palette (the absolute value of GPAL) in the blotch file BFIL will be used. If "0" is entered, the entire image area (512x512) will be used and BFIL will be ignored.

(6) **BFIL** is the name of the blotch file which defines the image area(s) of interest unless GPAL= 0. Only blotches defined by the graphics palette corresponding to GPAL will be used. The same blotch file used in creating the IFILs should be used in here.

DYNAMIC PARAMETERS:

I. Used when function key F1 is pressed.

(1) **OFIL** is the output file to create for saving the displayed image. The output file is a regular PC-SEAPAK image file (one header block followed by 512 blocks of image data).

II. Used when function key F4 is pressed.

(1) **XMIN** defines the lower end of the range of pixel values that will be rescaled for the displayed image. Pixel values which are less than or equal to XMIN will be set to "black" with gray level value 0. The value entered should conform to the units of the image (i.e., gray levels or other linear scales). The current image minimum is used as the default value of XMIN.

(2) **XMAX** defines the upper end of the range of pixel values that will be rescaled for the displayed image. Pixel values which are greater than or equal to XMAX will be set to "white" with gray level value 255. Again, the value entered should conform to the units of the image. The current image maximum is used as the value of XMAX.

(3) **INVAL** has the same meaning as in PARAMETERS section.

III. Used when function key F9 is pressed.

(1) **RANGE** contains two values which define the range of pixel values to be displayed on the histogram. These two values should conform to the units of the image (i.e., gray levels or other linear scales). RANGE values outside the image's current minimum and maximum (the default values for RANGE) will be ignored.

FUNCTION KEY DEFINITIONS:

ESC: Exits the program.

F1: Allows the user to save the currently displayed image to a user specified file.

F2: Allows the user to switch to another frame buffer. A message will display on the screen to show which frame buffer is currently displayed when this key is pressed.

F3: Displays a list of the frame buffers and data file names currently used in this program.

F4: Allows the user to rescale the image when MODE=1. Three parameters are required to be entered XMIN, XMAX and INVAL. One may use function keys F7 (image min/max) and F8 (box min/max) and F9 (histogram) for help in choosing appropriate XMIN and XMAX values for rescaling.

F5: Increases the current box size by 2. The box size will be reset to 3 once it exceeds 31. The box is used for function key F8 to display the minimum value, the maximum value and the mean value inside current box. This function key is not available when MODE=2.

F6: Decreases the current pixel length of a box side by 2. This length will be reset to 31 if it is less than 3. The box is used for function key F8 to display the minimum value, the maximum value and the mean value inside the current box. This function key is not available when MODE=2.

F7: Displays the current min/max scaling for the displayed image as well as the original scaling. This function key is not available when MODE=2.

F8: Displays the min/max, mean and total valid pixels within the box shown on the MVP-AT. This function key is not available when MODE=2.

F9: Allows the user to display the histogram of the presently displayed image with the current scaling when MODE=1. The user is prompted for the parameter RANGE which consists of two values that defines the range for the pixel values to be displayed on the histogram.

F10: Toggles the blotch area(s) on/off when a blotch area(s) is used with the input file(s) (GPAL <> 0).

ALT F1: Toggles the function key menu display on/off.

MOUSE RIGHT BUTTON - Same as ALT F1.

MOUSE LEFT BUTTON - Same as F8 when MODE=1.

PROGRAM NAME: STRETCH
DATE: 10/28/91
MENU: LUTCOLOR

DESCRIPTION: The program STRETCH provides a capability to linearly stretch the contrast of images on the MVP-AT display. The mouse is used to control the stretching by modifying the range of the input and output gray levels. When the cursor is moved to the left or right, the intercept of the input-to-output gray-level relationship line is shifted left or right, respectively. When the cursor is moved up or down, the slope of the line increases or decreases, respectively.

PARAMETERS:
There are no parameters.

DYNAMIC PARAMETERS:
I. Used for saving the stretched image into a frame buffer.
(1) **FRMBUF** is the image frame buffer in which to save the stretched image.
II. Used for saving the stretched image into a file.
(1) **O_FIL** is the name of the file to create for saving the stretched image. If the displayed image has a header block, it will be used for the newly created file except that the calibration slope and intercept will be modified to reflect the stretching performed with this program.

FUNCTION KEY DEFINITIONS:
ESC: Exits the program.
F1: Toggles the pixel value/gray level graphics display on/off.
F2: Displays the next image frame buffer.
F3: Allows the user to specify a frame buffer in which to save the stretched image.
F4: Toggles the image data between positive (0 to 255) and negative (255 to 0) linear ramps. This key will reset the linear stretch to the original status.
F5: Allows the user to save the stretched image into a file.
ALT F1: Toggles the function key menu display on/off.
MOUSE RIGHT BUTTON - Same as ALT F1

PROGRAM NAME: TABLOAD
DATE: 10/28/91
MENU: LUTCOLOR

DESCRIPTION: The program TABLOAD loads a disk-resident look-up table (LUT) into the palette used by the frame buffer FRMBUF. By default, the palettes 11 to 14 are reserved for the look-up tables of frame buffers 0 to 3. The LUT files saved in programs TABSAV and PAINT can be used in this program.

PARAMETERS:
(1) **LUTFIL** is the input LUT file name. The LUT data for red, green and blue will be loaded into the palette used in the frame buffer specified in parameter FRMBUF.
(2) **FRMBUF** is the index number (0 to 3) of the frame buffer. The look-up table of the palette (11 to 14) the frame buffer used will be loaded from the input LUT file specified in parameter LUTFIL to this frame buffer.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: TABSAV
DATE: 10/28/91
MENU: LUTCOLOR

DESCRIPTION: The program TABSAV saves the look-up table of the palette used for the frame buffer specified in parameter FRMBUF. By default, the palettes (11 to 14) in the output look-up table are reserved for the look-up tables of frame buffers 0 to 3. To be consistent with the output file created in the program PAINT, the file created in this program will contain 322 bytes of blank information for the color boxes (used in the program PAINT), then the 768 bytes for the red, green and blue look-up tables (256 entries each). Since the output file created by this program has no information on the color boxes, it cannot be loaded in the program PAINT, but it can be loaded in program TABLOAD.

PARAMETERS:
(1) **LUTFILE** is the output file name which the look-up table of the palette used in the frame buffer specified in parameter FRMBUF is to be saved to.
(2) **FRMBUF** is the index number (0 to 3) of the frame buffer. The look-up table of the palette (11 to 14) currently used in this frame buffer will be saved into a file.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: THRES
DATE: 10/28/91
MENU: CZCSL2

DESCRIPTION: This process "flags" or reassigns a certain range of gray levels in an input image to a specified gray level. Options are provided to alter the range and the output or "flag" gray level. It is useful in determining the land/cloud (level 1, band 5) and cloud overlay (level 1, band 1) thresholds to be used in the level 2 generation programs. The program is especially useful when a band 5 image has thin clouds which should be excluded from the analysis by enabling the user to display the range of flagged gray levels. Note that the convention used for the LANCLD and CLOUD parameters in the level-2 programs is exclusive meaning that all values exceeding their input values will be flagged. The programs LANCLD and FLAGLC use the same convention. THRES uses an inclusive convention in the RANGE parameter. The program requires the original level-1 image be loaded in one frame buffer and the resultant image will be displayed in another frame buffer.

PARAMETERS:

(1) **IFRMBUF** is the frame buffer number (1, 2 or 3) in which the original image is loaded in. Do not use the overlay frame buffer 0.

(2) **OFRMBUF** is the frame buffer which will be displayed showing the output product. Do not use the overlay frame buffer 0 or the frame buffer IFRMBUF.

(3) **RANGE** refers to all values (inclusive) that are to be reassigned to a single value, GRAY. THRES supports only one continuous input RANGE at a time.

(4) **GRAY** refers to the gray level (0-255, one value) to which all values in RANGE will be reassigned.

FUNCTION KEY DEFINITIONS:

ESC: Exits the program.

F1: Allows the user to change the gray level range in parameter RANGE and the gray level to be flagged in parameter GRAY.

F2: Toggles between the input frame buffer and the output frame buffer.

ALT F1: Toggles the function key menu display on/off.

MOUSE RIGHT BUTTON - Same as ALT F1.

PROGRAM NAME: TP2DSK
DATE: 10/28/91
MENU: CZCSIN

DESCRIPTION: TP2DSK ingests CZCS level-1 scenes from tape into disk files containing the scenes at full-resolution. CZCS calibrated radiance tapes (CRTs) are described in Williams et al. (1985a). Up to six minutes (three scenes) of data can be ingested from a tape into each disk file. Two types of output files are generated, a navigation information file and the full-resolution image files. The navigation file will have the same root name as specified in FILENAME parameter, but with the extension ".ANC". The image files will have the name with the number 1 to N attached to the root name (not the file name extension) of FILENAME, where N is the number entered for the parameter BANDS which decides how many files will be created for bands from 1 to 6. Due to the storage requirement of these image files, one should check the amount of free disk space before executing the program. A rough estimate for a single band with a full two-minute scene is 2 Mbytes. Only the program WINDOW can be used to view the output image files created by this program.

PARAMETERS:
(1) **SCENE** refers to the number of the first scene on the tape to be read.
(2) **TOTAL** refers to the total number of consecutive scenes to be ingested (1 to 3).
(3) **BANDS** is the number of the output files to be created. A value of 1 to 6 should be entered. A value of 1 will only create a file for band 1 and a value of N (=2, 3, 4, 5, 6) will create N files for bands 1 to N.
(4) **FULLIMG** is the name of the output file name. Numbers from 1 to N are added to the file name root (not to the extension) for each channel where the N is the input of BANDS, for the 443, 520, 550, 670, 780 nm and 12 micron bands, respectively.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: TPAVHRR
DATE: 10/28/91
MENU: AVHRRIN

DESCRIPTION: This program ingests a TIROS satellite series AVHRR level-1b scene from a data tape in the format of those generated by NOAA/NESDIS/NCDC/SDSD. Several PC-SEAPAK images may be created as specified by the parameter OUTPUT:

- Percent albedo image for visible channel 1
- Percent albedo image for visible channel 2
- Brightness energy temperature (deg C) image for IR channel 3
- Brightness energy temperature (deg C) image for IR channel 4
- Brightness energy temperature (deg C) image for IR channel 5 (if any)
- Sea-surface temperature (SST) image (deg C)

The algorithms for generating the visible and IR product images are described in the NOAA Polar Orbiter Data Users Guide (Kidwell, 1991). The algorithm options for calculating the SST values are detailed in the help text of SST_EQN.

The tape file may be that of HRPT, LAC, or GAC data for any of the TIROS/NOAA series satellites. However, the data must be in packed format, with time incrementing, and be a full data set copy (as opposed to selective extract subsets where certain channels are selected). Kidwell (1991) also contains all the specifications for the file format.

A file containing various information about the processing performed by this program will also be generated if any output image is requested. This log file will have the extension ".AVH". See the help text for the parameter O_FIL for more information about the convention for naming the output files.

A nonlinear calibration correction is applied to channels 4 and 5 for NOAA-9, 10, and 11 (Weinreb et al., 1990). This correction will therefore also impact the SST output image. However, if the line reduction factor (REDFAC(2)) is greater than four, this correction will not be applied. The reason for this exception is that data required for this correction are located piecemeal on consecutive scan lines with the pattern repeating every five lines. Reduction factors greater than four cause problems with aliasing of the data resulting from the subsampling. That is, the same (essential) parts of the data pattern may be missed from each repetition.

A time-dependent correction is applied to channels 1 and 2 for NOAA-7, 9, and 11 to account for deterioration of sensor sensitivity (Kaufman and Holben, 1991).

PARAMETERS:

(1) **O_FIL** is the file name to use as the basis for the names of the output files created by this program. If all image files are created (see OUTPUT), and O_FIL = "AVHRR.IMG", the files created will be named as follows:
AVHRR1.IMG > Percent albedo image for visible channel 1

AVHRR2.IMG	>	Percent albedo image for visible channel 2
AVHRR3.IMG	>	Brightness energy temperature (deg C) image for IR channel 3
AVHRR4.IMG	>	Brightness energy temperature (deg C) image for IR channel 4
AVHRR5.IMG	>	Brightness energy temperature (deg C) image for IR channel 5 (if any)
AVHRR6.IMG	>	SST image (deg C)
AVHRR.AVH	>	AVHRR ingestion log file
AVHRR.CTL	>	Navigation control point file.

(2) **SCENE** is the number of the scene to be ingested. For example, the first scene on the tape is number 1, the second is number 2, and so on.

(3) **SCAN** is the index number for one of the following options:

1. The program will run using the specified WINDOW, REDFAC and SAT_NO values.
2. The program will scan the specified scene on the tape in order to determine its first and last scan line numbers. The scan line limits will then be displayed and default values set for WINDOW, REDFAC and SAT_NO. You will then be prompted for these parameters before proceeding with the scene's ingestion.
3. This is the same option as 2 except that the program will proceed with the ingestion using the defaults without prompting you for WINDOW, REDFAC and SAT_NO.

Options 2 and 3 are useful if you do not know the scan-line size of the scene in order to choose appropriate WINDOW and REDFAC values. For options 2 and 3, the WINDOW defaults for samples (data points along scan lines) will be WINDOW(1)=1 and WINDOW(3)=2048 for LAC data or WINDOW(3)=409 for GAC data; the horizontal (along scan) reduction factor, REDFAC(1), will be 4 for LAC data or 1 for GAC data. The defaults for the scan line WINDOW(2) and WINDOW(4) will be the first and last scan line numbers for the scene, respectively. The vertical (along orbital track) reduction factor, REDFAC(2), will be assigned according to the number of scan lines contained in the scene so as to fit all requested scan lines within the display with as few blank display lines as possible.

(4) **CLOUD** is the minimum percent albedo which represents clouds in the channel 1 image. Pixels whose channel 1 albedo values are greater than or equal to CLOUD will have their SST image values set to absolute white (if OUTPUT(6)="Y"). The value 100 should be used for night scenes when the channel 1 visible image is not very meaningful.

(5) **SST_EQN** specifies the index of an equation to use for calculating the sea-surface temperatures (SST) from AVHRR data. If SST_EQN=0, equation 2, 4, 5, 7, 8, 9 or 13 will be used depending upon the satellite and whether it is a day or nighttime scene. (For this purpose, the program considers ascending scenes as day scenes and descending scenes as night scenes.) Recommendations for the use of the equations with the corresponding satellite and flight direction are given

below. In each equation, T(n) is the brightness energy temperature (Kidwell, 1991, p.3-14) in degrees Kelvin for AVHRR channel n and sec(SZA) is the secant of the satellite zenith angle. (References in parentheses on the right are given below.) Note that, if OUTPUT(6)="N", specifying that no SST image be created, SST_EQN will be ignored. Otherwise, pixels for which SSTs will be calculated may be restricted by the value of the CLOUD input parameter.

```
 1. SST = T(3)*C(1) + T(4)*C(2) + T(5)*C(3) + C(4)
    where C(n) are the values of the input parameter COEFS
    (generalized equation).
 2. SST = 1.3826*T(3) - 0.31*T(4) - 291.26
    for NOAA-6, day or night (Bernstein, 1982, p.9461).
 3. SST = 1.5*(T(3)-273.15) - 0.44*(T(4)-273.15) + 1.12
    for NOAA-6, day or night (McClain, 1981, p.2).
 4. SST = 1.0346*T(4) + 2.5800*(T(4)-T(5)) - 283.21
    for NOAA-7, day (Strong and McClain, 1984, p.139).
 5. SST = 1.0527*T(4) + 2.6272*(T(4)-T(5)) - 288.22
    for NOAA-7, night (Barbieri et al., 1983, p.20).
 6. SST = 1.0170*T(4) + 0.9700*(T(3)-T(5)) - 276.58
    for NOAA-7, night (Strong and McClain, 1984, p.139).
 7. SST = 3.6569*T(4) - 2.6705*T(5) - 268.92
    for NOAA-9, day (McClain et al., 1985, p.11600).
 8. SST = 3.6535*T(4) - 2.6680*T(5) - 268.41
    for NOAA-9, night (McClain et al., 1985, p.11600).
 9. SST = 1.0155*T(4) + 2.5*(T(4)-T(5))
          + 0.73*(T(4)-T(5))*(sec(SZA)-1) - 277.99
    for NOAA-11, day (Kidwell, 1991).
10. SST = (T(4)-T(5)+0.789)*(0.19069*T(5)-49.16)/
          (0.20524*T(5)-0.17334*T(4)-6.78) + 0.92912*T(5)
          + 0.81*(T(4)-T(5))*(sec(SZA)-1) - 254.18
    for NOAA-11, day (Kidwell, 1991).
11. SST = (T(3)-T(5)+14.86)*(0.16835*T(4)-34.32)/
          (0.20524*T(5)-0.07747*T(3)-20.01)
          + 0.97120*T(4) + 1.87*(sec(SZA)-1) - 276.59
    for NOAA-11, night (Kidwell, 1991).
12. SST = (T(3)-T(4)-6.44)*(0.17079*T(4)-58.47)/
          (0.17334*T(4)-0.07747*T(3)-33.74)
          + 0.98530*T(4) + 1.97*(sec(SZA)-1) - 257.28
    for NOAA-11, night (Kidwell, 1991).
13. SST = (T(4)-T(5)+1.46)*(0.19596*T(5)-48.61)/
          (0.20524*T(5)-0.17334*T(4)-6.11) + 0.95476*T(5)
          + 0.98*(T(4)-T(5))*(sec(SZA)-1) - 263.84
    for NOAA-11, night (Kidwell, 1991).
```

(6) **COEFS** are the coefficients to use for the generalized equation for calculating sea-surface temperatures (SST). This parameter is used only if SST_EQN=1, specifying this equation which has the form

```
SST = T(3)*COEFS(1) + T(4)*COEFS(2) + T(5)*COEFS(3)
      + COEFS(4)
```

where T(n) is the brightness energy temperature (see Kidwell, 1991, p.3-14) in degrees Kelvin for AVHRR channel n. Note

that, if OUTPUT(6)="N", specifying that no SST image be created, COEFS will be ignored even when SST_EQN=1.

(7) **OUTPUT** allows the user to specify which SEAPAK output image to generate from the ingestion of the AVHRR level 1b tape scene. OUTPUT(n)="Y" will cause the image file corresponding to the index n to be created:
1. Percent albedo image for visible channel 1
2. Percent albedo image for visible channel 2
3. Brightness energy temperature (deg C) image for IR channel 3
4. Brightness energy temperature (deg C) image for IR channel 4
5. Brightness energy temperature (deg C) image for IR channel 5 (if any)
6. Sea-surface temperature (SST) image (deg C)

See the help text of O_FIL for more information about the convention for naming the output files. See the main help text for information on calibration corrections.

(8) **D2GSLOPE** is the slope for converting output image data values into gray levels using the following scaling equation:

gray level = (data - D2GINTCP) * D2GSLOPE

The index of D2GSLOPE corresponds to that of OUTPUT--if OUTPUT(n)="N", D2GSLOPE(n) will be ignored. For channels 1 and 2, data are in percent albedo; for all other output images, data are in degrees centigrade.

The minimum and maximum values for all requested output images will be printed in the log file. You can use these min/max values to help you choose D2GSLOPE and D2GINTCP values that maximize the contrast of desired features in the output images when rerunning this program. However, remember that visual comparisons may be misleading for images that are of the same data type but have different scalings.

(9) **D2GINTCP** is the intercept for converting output image data values into gray levels using the following scaling equation:

gray level = (data - D2GINTCP) * D2GSLOPE

The index of D2GINTCP corresponds to that of OUTPUT--if OUTPUT(n)="N", D2GINTCP(n) will be ignored. For channels 1 and 2, data are in percent albedo; for all other output images, data are in degrees centigrade.

The minimum and maximum values for all requested output images will be printed in the log file. You can use these min/max values to help you choose D2GSLOPE and D2GINTCP values that maximize the contrast of desired features in the output images when rerunning this program. However, remember that visual comparisons may be misleading for images that are of the same data type but have different scalings.

(10) **WINDOW** defines, in conjunction with REDFAC, the area of the tape scene to use for generating the SEAPAK images. WINDOW(1) and WINDOW(3) specify the positions of the first and last samples (data points along scan lines), respectively, to use from each scan line. WINDOW(2) and WINDOW(4) specify the first and last scan line numbers to ingest from the scene.

See the help text for SCAN for more information on the use of WINDOW.

Note that sample positions are numbered from the start of each scan. In scenes for which the satellite is ascending (flying south to north), sample 1 will be the easternmost sample of each scan; for descending scenes, it will be the westernmost sample. Also note that scan line numbers are those recorded with each scan line in the tape file of the scene. These may not start at 1 for a given scene depending on the specifications used for generating the tape file. However, scan line numbers are chronological, the lowest numbered scan line being the earliest in the scene.

To extract a certain area from an unfamiliar AVHRR scene, you may first ingest the entire scene with SCAN=2 or 3. The data limits within the output images and whether the satellite was ascending or descending will be indicated in the .AVH log file. These data limits are in terms of pixels and lines of an image display unit where pixel 1 is on the left and line 1 at the top. The first and last sample numbers and scan numbers and the sample and scan line reduction factors will also be found in the log file.

As an example, let's say that you have ingested this overview (entire) scene with a sample range of 1 to 2047 (LAC data) and scan line number range of 86 to 1487 with reduction factors of 4 for samples and 3 for scan lines. (Data limits would be 1 and 512 for pixels and 23 and 490 for lines.) We'll assume that, after displaying the overview, you wish to extract for greater detail a rectangular portion of data from it between pixels 100 to 150 and lines 130 to 200 (the display being 512 pixels by 512 lines).

The sample range (WINDOW(1) and WINDOW(3)) for the rectangle will be:

```
             (100)    *4 - (4-1) =  397 and
             (150)    *4 - (4-1) =  597 if descending
or     (512-150+1)*4 - (4-1) = 1449 and
       (512-100+1)*4 - (4-1) = 1649 if ascending
```

The scan line range (WINDOW(2) and WINDOW(4)) for the rectangle will be:

```
       (130- 23+1)*3 - (3-1) + (86-1) =  407 and
       (200- 23+1)*3 - (3-1) + (86-1) =  617 if descending
or     (490-200+1)*3 - (3-1) + (86-1) =  956 and
       (490-130+1)*3 - (3-1) + (86-1) = 1166 if ascending
```

The number of samples per scan line, 200, and the number of scan lines, 210, would then suggest reduction factors (REDFAC) of -2. The data in the resulting output images would be 400 pixels wide and 420 lines high.

(11) **REDFAC** are the reduction factors for the horizontal (along scan) and vertical (along orbital track) directions, in that order. Positive values indicate reduction by subsampling whereas negative values indicate magnification by pixel replication. For example, an entry of (2,2) will create images half as wide in samples and half as high in scan lines

as the scene area defined by WINDOW; an entry of (-2,-2) will generate images twice as high and wide. Note that reduction in this sense indicates an increase in geographical coverage while expansion indicates a decrease. Values of -1, 0, or 1 are equivalent and generate images having a one-to-one correspondence of pixels with the tape scene. See the help text for SCAN for more information on the use of REDFAC.

(12) **SAT_NO** is the NOAA series' satellite number which is normally encoded in the file header of the scene to be ingested as described in the NOAA Polar Orbiter Data Users Guide (Kidwell, 1991). The default value of SAT_NO (0) causes the program to use this header code to determine the satellite number. If the code does not correspond to the standard NOAA code or it is otherwise incorrect, you may enter the actual satellite number for SAT_NO.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: TPCZCS
DATE: 10/28/91
MENU: CZCSIN

DESCRIPTION: TPCZCS is used to ingest CZCS level-1 tape data into image files of 512x512 pixels. CZCS calibrated radiance tapes (CRTs) are described in Williams et al. (1985a). For multichannel data, an image file is generated for each channel with the channel number added at the end of the file name. The program also creates an additional control point file for navigation having the extension ".CTL". Each image file contains 513 blocks (each block is 512 bytes), the first block being header information which may be viewed using DMPHDR.

PARAMETERS:

(1) **O_FIL** is the file name to use as the basis for the names of the output files created by this program. If O_FIL = "CZCSLV1.IMG", the files created will be named as follows:

```
CZCSLV11.IMG > For band 1 (430 nm)
CZCSLV12.IMG > For band 2 (520 nm)
CZCSLV13.IMG > For band 3 (550 nm)
CZCSLV14.IMG > For band 4 (670 nm)
CZCSLV15.IMG > For band 5 (780 nm)
CZCSLV16.IMG > For band 6 (12 microns)
CZCSLV1.CTL  > Navigation control point file.
```

(2) **SCENE** refers to the scene number on the tape to be ingested. For CZCS level-1 tape, each scene has two files and only the second file will be used.

(3) **WINDOW** defines, in conjunction with REDFAC, the area of the tape scene to use for generating the PC-SEAPAK images. WINDOW(1) and WINDOW(3) specify the positions of the first and last samples (data points along scan lines), respectively, to use from each scan line. Note that sample positions are numbered from the start of each scan. Since CZCS data were only collected while the satellite was ascending (flying south to north), sample 1 will be the westernmost sample of each scan. WINDOW(2) and WINDOW(4) specify the first and last scan line numbers to ingest from the scene. CZCS level-1 scenes have a maximum of 970 scan lines corresponding to two minutes of data and a fixed number of 1968 pixels per scan line. Since TPCZCS generates only 512x512 pixel image files, subsampling is required to generate an overview of the scene. Blank pixels on the side margins and lines at the bottom of an image are added if the data do not fill the 512x512 array. An error message is generated if an improper combination of reduction factors and window values is input.

(4) **REDFAC** is the reduction factors for the horizontal (along scan) and vertical (along orbital track) directions, in that order. Positive values indicate reduction by subsampling whereas negative values indicate magnification by pixel replication. For example, an entry of (2,2) will create

images half as wide in samples and half as high in scan lines as the scene area defined by WINDOW; an entry of (-2,-2) will generate images twice as high and wide. Note that reduction in this sense indicates an increase in geographical coverage while expansion indicates a decrease. Values of -1, 0, or 1 are equivalent and generate images having a one-to-one correspondence of pixels with the tape scene.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: TPSDRPS
DATE: 10/28/91
MENU: AVHRRIN

DESCRIPTION: This program ingests a TIROS satellite series AVHRR level 1b scene from a data tape in the format of those generated by NORDA/SDRPS. Several PC-SEAPAK images may be created as specified by the parameter OUTPUT:

- Percent albedo image for visible channel 1
- Percent albedo image for visible channel 2
- Brightness energy temperature (deg C) image for IR channel 3
- Brightness energy temperature (deg C) image for IR channel 4
- Brightness energy temperature (deg C) image for IR channel 5 (if any)
- Sea-surface temperature (SST) image (deg C)

The algorithms for generating the visible and IR product images are described in the NOAA Polar Orbiter Data Users Guide (Kidwell, 1991). The algorithm options for calculating the SST values are detailed in the help text of SST_EQN.

The tape file may be that of HRPT, LAC, or GAC data for any of the TIROS/NOAA series satellites. However, the data must be in packed format, with time incrementing, and be a full data set copy (as opposed to selective extract subsets where certain channels are selected). The NOAA Polar Orbiter Data Users Guide also contains all the specifications for the file format. The input parameter TBM allows the program to handle two types of tape data - with or without the dummy TBM file for each scene in the tape.

A file containing various information about the processing performed by this program will also be generated if any output image is requested. This log file will have the extension ".AVH". See the help text for the parameter O_FIL for more information about the convention for naming the output files.

A nonlinear calibration correction is applied to channels 4 and 5 for NOAA-9, 10, and 11 (Weinreb et al., 1990). This correction will therefore also impact the SST output image. However, if the line reduction factor (REDFAC(2)) is greater than four, this correction will not be applied. The reason for this exception is that data required for this correction are located piecemeal on consecutive scan lines with the pattern repeating every five lines. Reduction factors greater than four cause problems with aliasing of the data resulting from the subsampling. That is, the same (essential) parts of the data pattern may be missed from each repetition.

A time-dependent correction is applied to channels 1 and 2 for NOAA-7, 9, and 11 to account for deterioration of sensor sensitivity (Kaufman and Holben, 1991).

PARAMETERS:

(1) **O_FIL** is the file name to use as the basis for the names of the output files created by this program. If all image files

are created (see OUTPUT), and O_FIL = "AVHRR.IMG", the files created will be named as follows:

AVHRR1.IMG	>	Percent albedo image for visible channel 1
AVHRR2.IMG	>	Percent albedo image for visible channel 2
AVHRR3.IMG	>	Brightness energy temperature (deg C) image for IR channel 3
AVHRR4.IMG	>	Brightness energy temperature (deg C) image for IR channel 4
AVHRR5.IMG	>	Brightness energy temperature (deg C) image for IR channel 5 (if any)
AVHRR6.IMG	>	SST image (deg C)
AVHRR.AVH	>	AVHRR ingestion log file
AVHRR.CTL	>	Navigation control point file.

(2) **TBM** is a flag to indicate whether there is a dummy TBM file for each scene in the tape. A "0" means there is no dummy TBM file and a "1" means there is a dummy TBM file for each scene in the tape.

(3) **SCENE** is the number of the scene to be ingested. For example, the first scene on the tape is number 1, the second is number 2, and so on.

(4) **SCAN** is the index number for one of the following options:

1. The program will run using the specified WINDOW, REDFAC and SAT_NO values.
2. The program will scan the specified scene on the tape in order to determine its first and last scan line numbers. The scan line limits will then be displayed and default values set for WINDOW, REDFAC and SAT_NO. You will then be prompted for these parameters before proceeding with the scene's ingestion.
3. This is the same option as 2 except that the program will proceed with the ingestion using the defaults without prompting you for WINDOW, REDFAC and SAT_NO.

Options 2 and 3 are useful if you do not know the scan-line size of the scene in order to choose appropriate WINDOW and REDFAC values. For options 2 and 3, the WINDOW defaults for samples (data points along scan lines) will be WINDOW(1)=1 and WINDOW(3)=2048 for LAC data or WINDOW(3)=409 for GAC data; the horizontal (along scan) reduction factor, REDFAC(1), will be 4 for LAC data or 1 for GAC data. The defaults for the scan line WINDOW(2) and WINDOW(4) will be the first and last scan line numbers for the scene, respectively. The vertical (along orbital track) reduction factor, REDFAC(2), will be assigned according to the number of scan lines contained in the scene so as to fit all requested scan lines within the display with as few blank display lines as possible.

(5) **OUTPUT** allows the user to specify which SEAPAK output image to generate from the ingestion of the AVHRR level 1b tape scene. OUTPUT(n)="Y" will cause the image file corresponding to the index n to be created:

1. Percent albedo image for visible channel 1
2. Percent albedo image for visible channel 2

3. Brightness energy temperature (deg C) image for IR channel 3
4. Brightness energy temperature (deg C) image for IR channel 4
5. Brightness energy temperature (deg C) image for IR channel 5 (if any)
6. Sea-surface temperature (SST) image (deg C)

See the help text of O_FIL for more information about the convention for naming the output files.

(6) **D2GSLOPE** is the slope for converting output image data values into gray levels using the following scaling equation:

gray level = (data - D2GINTCP) * D2GSLOPE

The index of D2GSLOPE corresponds to that of OUTPUT--if OUTPUT(n)="N", D2GSLOPE(n) will be ignored. For channels 1 and 2, data are in percent albedo; for all other output images, data are in degrees centigrade.

The minimum and maximum values for all requested output images will be printed in the log file. You can use these min/max values to help you choose D2GSLOPE and D2GINTCP values that maximize the contrast of desired features in the output images when rerunning this program. However, remember that visual comparisons may be misleading for images that are of the same data type but have different scalings.

(7) **D2GINTCP** is the intercept for converting output image data values into gray levels using the following scaling equation:

gray level = (data - D2GINTCP) * D2GSLOPE

The index of D2GINTCP corresponds to that of OUTPUT--if OUTPUT(n)="N", D2GINTCP(n) will be ignored. For channels 1 and 2, data are in percent albedo; for all other output images, data are in degrees centigrade.

The minimum and maximum values for all requested output images will be printed in the log file. You can use these min/max values to help you choose D2GSLOPE and D2GINTCP values that maximize the contrast of desired features in the output images when rerunning this program. However, remember that visual comparisons may be misleading for images that are of the same data type but have different scalings.

(8) **CLOUD** is the minimum percent albedo which represents clouds in the channel 1 image. Pixels whose channel 1 albedo values are greater than or equal to CLOUD will have their SST image values set to absolute white (if OUTPUT(6)="Y"). The value 100 should be used for night scenes when the channel 1 visible image is not very meaningful.

(9) **SST_EQN** specifies the index of an equation to use for calculating the sea-surface temperatures (SST) from AVHRR data. If SST_EQN=0, equation 2, 4, 5, 7, 8, 9 or 13 will be used depending upon the satellite and whether it is a day or nighttime scene. (For this purpose, the program considers ascending scenes as day scenes and descending scenes as night scenes.) Recommendations for the use of the equations with the corresponding satellite and flight direction are given below. In each equation, T(n) is the brightness energy

temperature (Kidwell, 1991, p.3-14) in degrees Kelvin for AVHRR channel n and sec(SZA) is the secant of the satellite zenith angle. (References in parentheses on the right are given below.) Note that, if OUTPUT(6)="N", specifying that no SST image be created, SST_EQN will be ignored. Otherwise, pixels for which SSTs will be calculated may be restricted by the value of the CLOUD input parameter.

```
 1. SST = T(3)*C(1) + T(4)*C(2) + T(5)*C(3) + C(4)
    where C(n) are the values of the input parameter COEFS
    (generalized equation).
 2. SST = 1.3826*T(3) - 0.31*T(4) - 291.26
    for NOAA-6, day or night (Bernstein, 1982, p.9461).
 3. SST = 1.5*(T(3)-273.15) - 0.44*(T(4)-273.15) + 1.12
    for NOAA-6, day or night (McClain, 1981, p.2).
 4. SST = 1.0346*T(4) + 2.5800*(T(4)-T(5)) - 283.21
    for NOAA-7, day (Strong and McClain, 1984, p.139).
 5. SST = 1.0527*T(4) + 2.6272*(T(4)-T(5)) - 288.22
    for NOAA-7, night (Barbieri et al., 1983, p.20).
 6. SST = 1.0170*T(4) + 0.9700*(T(3)-T(5)) - 276.58
    for NOAA-7, night (Strong and McClain, 1984, p.139).
 7. SST = 3.6569*T(4) - 2.6705*T(5) - 268.92
    for NOAA-9, day (McClain et al., 1985, p.11600).
 8. SST = 3.6535*T(4) - 2.6680*T(5) - 268.41
    for NOAA-9, night (McClain et al., 1985, p.11600).
 9. SST = 1.0155*T(4) + 2.5*(T(4)-T(5))
          + 0.73*(T(4)-T(5))*(sec(SZA)-1) - 277.99
    for NOAA-11, day (Kidwell, 1991).
10. SST = (T(4)-T(5)+0.789)*(0.19069*T(5)-49.16)/
          (0.20524*T(5)-0.17334*T(4)-6.78) + 0.92912*T(5)
          + 0.81*(T(4)-T(5))*(sec(SZA)-1) - 254.18
    for NOAA-11, day (Kidwell, 1991).
11. SST = (T(3)-T(5)+14.86)*(0.16835*T(4)-34.32)/
          (0.20524*T(5)-0.07747*T(3)-20.01)
          + 0.97120*T(4) + 1.87*(sec(SZA)-1) - 276.59
    for NOAA-11, night (Kidwell, 1991).
12. SST = (T(3)-T(4)-6.44)*(0.17079*T(4)-58.47)/
          (0.17334*T(4)-0.07747*T(3)-33.74)
          + 0.98530*T(4) + 1.97*(sec(SZA)-1) - 257.28
    for NOAA-11, night (Kidwell, 1991).
13. SST = (T(4)-T(5)+1.46)*(0.19596*T(5)-48.61)/
          (0.20524*T(5)-0.17334*T(4)-6.11) + 0.95476*T(5)
          + 0.98*(T(4)-T(5))*(sec(SZA)-1) - 263.84
    for NOAA-11, night (Kidwell, 1991).
```

(10) **COEFS** are the coefficients to use for the generalized equation for calculating sea-surface temperatures (SST). This parameter is used only if SST_EQN=1, specifying this equation which has the form

```
SST = T(3)*COEFS(1) + T(4)*COEFS(2) + T(5)*COEFS(3)
      + COEFS(4)
```

where T(n) is the brightness energy temperature (see Kidwell, 1991, p.3-14) in degrees Kelvin for AVHRR channel n. Note

that, if OUTPUT(6)="N", specifying that no SST image be created, COEFS will be ignored even when SST_EQN=1.

(11) **WINDOW** defines, in conjunction with REDFAC, the area of the tape scene to use for generating the SEAPAK images. WINDOW(1) and WINDOW(3) specify the positions of the first and last samples (data points along scan lines), respectively, to use from each scan line. WINDOW(2) and WINDOW(4) specify the first and last scan line numbers to ingest from the scene. See the help text for SCAN for more information on the use of WINDOW.

Note that sample positions are numbered from the start of each scan. In scenes for which the satellite is ascending (flying south to north), sample 1 will be the easternmost sample of each scan; for descending scenes, it will be the westernmost sample. Also note that scan line numbers are those recorded with each scan line in the tape file of the scene. These may not start at 1 for a given scene depending on the specifications used for generating the tape file. However, scan line numbers are chronological, the lowest numbered scan line being the earliest in the scene.

To extract a certain area from an unfamiliar AVHRR scene, you may first ingest the entire scene with SCAN=2 or 3. The data limits within the output images and whether the satellite was ascending or descending will be indicated in the .AVH log file. These data limits are in terms of pixels and lines of an image display unit where pixel 1 is on the left and line 1 at the top. The first and last sample numbers and scan numbers and the sample and scan line reduction factors will also be found in the log file.

As an example, let's say that you have ingested this overview (entire) scene with a sample range of 1 to 2047 (LAC data) and scan line number range of 86 to 1487 with reduction factors of 4 for samples and 3 for scan lines. (Data limits would be 1 and 512 for pixels and 23 and 490 for lines.) We'll assume that, after displaying the overview, you wish to extract for greater detail a rectangular portion of data from it between pixels 100 to 150 and lines 130 to 200 (the display being 512 pixels by 512 lines).

The sample range (WINDOW(1) and WINDOW(3)) for the rectangle will be:

```
                (100)   *4 - (4-1) =  397 and
                (150)   *4 - (4-1) =  597 if descending
or        (512-150+1)*4 - (4-1) = 1449 and
          (512-100+1)*4 - (4-1) = 1649 if ascending
```

The scan line range (WINDOW(2) and WINDOW(4)) for the rectangle will be:

```
       (130- 23+1)*3 - (3-1) + (86-1) =  407 and
       (200- 23+1)*3 - (3-1) + (86-1) =  617 if descending
or     (490-200+1)*3 - (3-1) + (86-1) =  956 and
       (490-130+1)*3 - (3-1) + (86-1) = 1166 if ascending
```

The number of samples per scan line, 200, and the number of scan lines, 210, would then suggest reduction factors (REDFAC)

of -2. The data in the resulting output images would be 400 pixels wide and 420 lines high.

(12) **REDFAC** are the reduction factors for the horizontal (along scan) and vertical (along orbital track) directions, in that order. Positive values indicate reduction by subsampling whereas negative values indicate magnification by pixel replication. For example, an entry of (2,2) will create images half as wide in samples and half as high in scan lines as the scene area defined by WINDOW; an entry of (-2,-2) will generate images twice as high and wide. Note that reduction in this sense indicates an increase in geographical coverage while expansion indicates a decrease. Values of -1, 0, or 1 are equivalent and generate images having a one-to-one correspondence of pixels with the tape scene. See the help text for SCAN for more information on the use of REDFAC.

(13) **SAT_NO** is the NOAA series' satellite number which is normally encoded in the file header of the scene to be ingested as described in the NOAA Polar Orbiter Data Users Guide (Kidwell, 1991). The default value of SAT_NO (0) causes the program to use this header code to determine the satellite number. If the code does not correspond to the standard NOAA code or it is otherwise incorrect, you may enter the actual satellite number for SAT_NO.

FUNCTION KEY DEFINITIONS:
No function keys are used.

PROGRAM NAME: TRACK
DATE: 10/28/91
MENU: OVERLAYS

DESCRIPTION: This program overlays a ship/aircraft track from an ASCII file onto a displayed image and allows retrieval of the track data as well as the image data along that track. Different segments of the ship/aircraft track (or the full track) can be statistically compared with the image data and plotted. The values for both ship/aircraft track and image data can be output to a file or to the printer. Other options allow the user to zoom and roam the displayed image and the ship track, to shift the track, to select a new parameter (data) in the track file, and to change the data type for the displayed image.

PARAMETERS:

(1) **FRMBUF** is the frame buffer that contains the image to be overlaid with the ship track.

(2) **F_TRACK** is the input ASCII ship track file. This file must contain at least two columns of latitude and longitude of track data and optionally up to eight other data columns. The order of the latitude and longitude and data columns can be random, but must be consistent for all records. The parameters PAR_LAT, PAR_LON and PAR_VAR are used to select those columns when read in from this file. Note that all the data will be read in as real values by Fortran free format.

(3) **PAR_LAT** is the number of the data column in file F_TRACK that contains the latitude locations of the ship track.

(4) **PAR_LON** is the number of the data column in file F_TRACK that contains the longitude locations of the ship track.

(5) **PAR_VAR** is the number of the data column in file F_TRACK that contains the ship data to be examined, compared and plotted. A 0 should be entered if there are no ship data to be retrieved.

(6) **PAR_RNG** is the minimum and maximum range for the ship track data to do statistical calculations with the image data. The default value "-9999." means two extreme values will be used and all the data will be considered as valid values. For the image data, the gray level value is used initially and (0, 255) is used as the range for valid pixels. However, the function key F7 allows the user to change the image data type as well as the valid range and the function key F8 allows the user to change the ranges for both the ship track data and the image data.

(7) **PAR_TIT** is the title for the ship track data that will be used for the output and as the default label for the plot.

(8) **LN_SKIP** is the number of lines to be skipped in the F_TRACK file before reading the ship track data.

(9) **GPAL** is the graphics palette for the ship track.

(10) **LAT_SI** is the slope and intercept to rescale the input latitude data. The latitude data read in from F_TRACK will be rescaled by

LAT (in TRACK) = LAT (in F_TRACK) * SLOPE + INTERCEPT

before used in this program.

(11) **LON_SI** is the slope and intercept to rescale the input longitude data. The longitude data read in from F_TRACK will be rescaled by

LON (in TRACK) = LON (in F_TRACK) * SLOPE + INTERCEPT

before used in this program.

(12) **VAR_SI** is the slope and intercept to rescale the input ship track data. The ship track data read in from F_TRACK will be rescaled by

DATA (in TRACK) = DATA (in F_TRACK) * SLOPE + INTERCEPT

before used in this program.

DYNAMIC PARAMETERS:

I. Parameters for generating plots:

(1) **GPAL1** is the graphics palette for the ship track data plot.

(2) **MARK1** decides the symbol to be used for the ship track data plot. The value "1" will use "x" and the value "2" will use "+" for the plot.

(3) **RANGE1** is the minimum and maximum range for the ship data to be plotted. By default, the minimum and maximum values of the ship data will be used each time the F4 key is pressed.

(4) **LABEL1** is the ship track data title to be displayed in the plot. If OPTION is 1 (combined plot) the title will be displayed on top of the plot, otherwise, OPTION is 2 for separated plots, the title will be displayed on the left side of the Y-axis.

(5) **GPAL2** is the graphics palette for the image data plot.

(6) **MARK2** decides the symbol to be used for the image data plot. The value "1" will use "x" and the value "2" will use "+" for the plot.

(7) **RANGE2** is the minimum and maximum range for the image data to be plotted. By default, the minimum and maximum values of the ship data will be used each time the F4 key is pressed.

(8) **LABEL2** is the image data title to be displayed in the plot. If OPTION is "1" (combined plot) the title will be displayed on top of the plot, otherwise, OPTION is "2" (separated plots), the title will be displayed on the left side of the Y-axis.

(9) **OPTION** selects to combine or to separate the plots of ship data and image data. The value "1" will plot the ship data and the image data in the same coordinate. The value "2" will plot the ship data and the image data in two different coordinates.

II. Parameters for changing the image data type:

(1) **MODE** specifies the data type of the displayed image. A value of "1" (the default value) should be entered if the pixel values of the displayed image represent data (such as temperature) that are linearly related to gray levels. A

value of "2" should be entered if they represent pigment concentrations (mg/m3).

(2) **FACTOR** is a non-negative scaling factor that is used only if MODE=1, i.e. the data-to-gray scale mapping function is linear for the displayed image. It is ignored when MODE=2. If FACTOR is positive, it will represent the factor by which to divide the gray values in order to convert them into actual data values. If zero is entered, the slope and intercept for this mapping function will be obtained from the header of the disk file for the displayed image. In order to retain the gray values, a "1" (the default value) should be entered; for sea surface temperature (SST), enter 8; for water radiance data, enter 85; for aerosol radiance data, enter 100.

(3) **IMG_RNG** defines the range of the image data to use for the statistical calculation with the ship data. Two values should be entered that conform to the units of the displayed image as specified by MODE and FACTOR.

(4) **IMG_TIT** is the title for the image data which will be used for the output and as the default label for the plot.

III. Parameters for changing the ship and image data ranges:

(1) **PAR_RNG** is the minimum and maximum range for the ship track data to do statistical calculations with the image data. The default value "-9999." means two extreme values will be used and all the data will be considered as valid values. For the image data, the gray level value is used initially and (0, 255) is used as the range for valid pixels. However, the function key F7 allows the user to change the image data type as well as the valid range and the function key F8 allows the user to change the ranges for both the ship track data and the image data.

(2) **PAR_TIT** is the title for the ship track data which will be used for the output and as the default label for the plot.

(3) **IMG_RNG** defines the range of the image data to use for the statistical calculations with the ship data. Two values should be entered that conform to the units of the displayed image as specified by MODE and FACTOR.

(4) **IMG_TIT** is the title for the image data that will be used for the output and as the default label for the plot.

IV. Parameters for shifting ship track:

(1) **X_SHIFT** is the number of pixels to shift the ship track. Positive value shifts to the right and negative value shifts to the left.

(2) **Y_SHIFT** is the number of lines to shift the ship track. Positive value shifts down and negative value shift up.

FUNCTION KEY DEFINITIONS:

ESC: Exits the program TRACK.

F1: Outputs the mean and the standard deviation of the ship track data and the image data as well as their RMS (Root Mean Square), covariance and correlation coefficient. The pixel and line locations of the ship track, the ship track data and the corresponding image data will also be outputted. The user

may define any two points segment on the ship track, by using F5 and F6 keys, for this output.

F2: Allows the user to output the data discussed in F1 key to a file or a printer.

F3: Allows the user to change all the parameters specified in the PARAMETERS section.

F4: Allows the user to plot the ship track data and the image data. The parameters specified in DYNAMIC PARAMETERS I will be asked.

F5: Defines the first point of a segment on the ship track. The cursor has to be on the ship track for this key to define the first point. Once a point is defined, it will be flashed on/off on the display. To redefine the first point, the user needs to press this key again. By default, the whole track will be considered as a segment. Keys F5 and F6 allow the user to define any segment of the whole track for the outputs and plots.

F6: Defines the second point of a segment on the ship track. The cursor has to be on the ship track for this key to define the second point. Once a point is defined, it will be flashed on/off on the displayed. To redefine the second point, the user just needs to press this key again. By default, the whole track will be considered as a segment. Key F5 and key F6 allow the user to define any segment of the whole track for the outputs and plots.

F7: Allows the user to convert the image gray level data into pigment data or other linear scaled geophysical data. The parameters specified in DYNAMIC PARAMETERS II will be asked.

F8: Allows the user to change the valid ranges for the ship data and the image data for statistical calculation as well as their titles for the output. The parameters specified in DYNAMIC PARAMETERS III will be asked.

F9: Allows the user to shift the ship track horizontally and/or vertically. The parameters specified in DYNAMIC PARAMETERS IV will be asked.

F10: Allows the user to roam and zoom the displayed image and the overlay of the ship track. Once this key is pressed, the mouse left button is used to zoom in the top-left corner of the displayed image and overlay graphics. The mouse right button is used to toggle the control of the mouse movement for the cursor or for the image roaming. Press this key again to restore the image to the original display.

ALT F1: Toggles function key menu display on/off.
ALT F5: Clears the first point defined by key F5.
ALT F6: Clears the second point defined by key F6.
ALT F7: Turns the ship track on/off.
ALT F8: Clears the ship data and the image data plots.
ALT F9: Turns the displayed image on/off.
MOUSE RIGHT BUTTON : Toggles function key menu display on/off.

PROGRAM NAME: TSERIES
DATE: 10/28/91
MENU: STAT1

DESCRIPTION: This program determines the mean, standard deviation and/or coefficient of determination for a time series sequence of images. These data can then be plotted on the MVP-AT image display monitor when the program is running in the interactive mode. Only pixels within the specified blotched regions and data values within the specified data range are processed. In the interactive mode, options are provided to support the rescaling of the vertical axis, redefining of the blotch regions (including saving and loading blotch regions to and from a file), redefining the data range, toggling on/off the displayed image and the blotch regions, and outputting an ASCII file of the plot data. Note that under the interactive mode, the MVP-AT image frame buffer 1 and 2 will always be used for the image display and the temporary buffer for the blotch and any image or graphics on these two buffers will be destroyed. This program is also designed so that it can be run in a batch (non-interactive) mode which uses a blotch file instead of the blotch regions created under the interactive mode. In this case, a blotch file and an output file must be given.

PARAMETERS:

(1) **IMGFILS** are the file names of the input time series. Up to 36 image file names may be entered. However, since the wild card (* or ?) file format is supported, up to 300 image files can be processed. Note that if there is only one file name entered for IMGFILS, the program will assume it is a text file and read as input the time series whose names are listed in this file. Note that all the image files should have a header block.

(2) **RUNMODE** specifies interactive or batch mode of operation. If 0 is entered, batch mode is used, otherwise interactive mode is used. Note that in batch mode the parameters BLOFIL and the OUTFIL cannot be blank.

(3) **BLOFIL** is the name of the blotch file and is used only in the batch processing mode (RUNMODE = 0). The blotch file name (created by the program BLOTCH or BPSAV) is used together with the blotch graphics palette (parameter B_PAL) to define the regions in the input image files to be processed. For interactive processing (RUNMODE = 1), the blotched regions should be previously defined (either by BLOTCH or by BPLOAD) and displayed on the MVP-AT display monitor when this program is invoked.

(4) **OUTFIL** is the file name for outputting statistics. This file name is used for the output save file for the batch processing mode only. It will contain the orbit or scene number, the valid number of pixels, mean, standard deviation and coefficient of determination for each input image file.

(5) **MODE** specifies whether to calculate the statistics for the output images based on gray level values or to convert first to pigment concentration values. One should enter a "1" for gray level or a "2" for pigment concentration.

(6) **BPAL** is the number of a graphics palette which determines the color of the blotched regions to be used. A value in the range of 1 to 7 should be entered. In the batch processing mode, this parameter together with the blotch file BLOFIL define the processing regions. Only the regions in the blotch file which have the value of BPAL are processed. In the interactive processing mode, only the regions which have the same color as the graphics palette BPAL defined will be processed. The user can use program BLOTCH or BPSAV to display a blotched region before running this program, or use the function key F10 in this program to generate or modify any blotched regions.

(7) **ORBIT1** defines the lowest orbit number to be used in the processing.

(8) **ORBIT2** defines the highest orbit number to be used in the processing.

(9) **MINRNG** is the minimum range for valid pixel identification. This value and MAXRNG should conform to the data units specified by the parameter MODE. Only pixels within the MINRNG and MAXRNG range are used in the statistical calculations. Land/clouds and extremely high/low data pixels can be easily excluded from the statistical calculations by entering proper values of MINRNG and MAXRNG.

(10) **MAXRNG** is the maximum range for valid pixel identification. This value and MINRNG should conform to the data units specified by the parameter MODE. Only pixels within the MINRNG and MAXRNG range are used in the statistical calculations. Land/clouds and extremely high/low data pixels can be easily excluded from the statistical calculations by entering proper values of MINRNG and MAXRNG.

(11) **VALID** is the minimum number of valid pixels required before any plot of the statistics will be generated. A valid pixel is a pixel that is in the specified blotched region and is within the specified data range.

DYNAMIC PARAMETERS: (only for interactive mode)

I. Used for changing the graphics palettes for blotch and plot.

(1) **BPAL** is the graphics palette that determines the color of the blotched regions to use for calculating the statistics data. Once the value has been changed, all the time series image files will be redisplayed one by one to get the new statistics data.

(2) **PL_PAL** is the graphics palette to use for the plot.

(3) **AX_PAL** is the graphics palette to use for the X and Y axes and the tick marks and tick labels on the axes.

(4) **LB_PAL** is the graphics palette to use for the labels of the X and Y axes.

II. Used for changing the data mode and the data range.

(1) **MODE** is the same as MODE in the parameter section.
(2) **MINRNG** is the same as MINRNG in the parameter section.
(3) **MAXRNG** is the same as MAXRNG in the parameter section.
III. Used for changing the minimum number of valid pixels.
(1) **VALID** is the minimum number of valid pixels required before any plot of the statistics will be generated. A valid pixel is a pixel that is in the specified blotched region and is within the specified data range.
IV. Used in changing the plot range on the Y axis.
(1) **MIN_MEAN** is the minimum mean value to be plotted.
(2) **MAX_MEAN** is the maximum mean value to be plotted.
(3) **MIN_STD** is the minimum value of the standard deviation to be plotted.
(4) **MAX_STD** is the maximum value of the standard deviation to be plotted.
(5) **MIN_CD** is the minimum value of the coefficient of determination to be plotted.
(6) **MAX_CD** is the maximum value of the coefficient of determination to be plotted.
V. Used for getting the output file name.
(1) **OUTFIL1** is the file name to use for the output. If the user wants to send the output to the printer, "LPT1" or "LPT2" should be used as the value of OUTFIL1.

FUNCTION KEY DEFINITIONS:
ESC: Exits the program.
F1: Generates a plot of the mean versus the orbit number or the scene number overlaid on the displayed image. The orbit number will be used only when the number of input image files is less than 11 and all have orbit numbers. Otherwise, the scene number will be used.
F2: Generates a plot of the standard deviation versus the orbit number or the scene number overlaid on the displayed image. The orbit number will be used only when the number of input image files is less than 11 and all have orbit numbers. Otherwise, the scene number will be used.
F3: Generates a plot of the coefficient of determination versus the orbit number or the scene number overlaid on the displayed image. The orbit number will be used only when the number of the input image files are less than 11 and all have orbit numbers. Otherwise, the scene number will be used.
F4: Changes the graphics palettes used to define the blotch regions and to draw the plot, the axes, the tick marks, the tick labels and the axes labels. The parameters BPAL, PL_PAL, AX_PAL and LB_PAL will be requested if this key is pressed.
F5: Selects a new data mode and new range of valid pixels. The parameters MODE, MINRNG and MAXRNG will be requested if this key is pressed. Any changes in MODE, MINRNG and MAXRNG will cause all the input images files to be redisplayed one by one to get new statistics data.

F6: Redefines the minimum number of valid pixels required before any plots of the statistics will be generated. The parameter VALID will be requested if this key is pressed.

F7: Redefines the data range of the mean, standard deviation or the coefficient of determination to be plotted. Depending on the current plot mode, either the MIN_MEAN, MAX_MEAN or the MIN_STD, MAX_STD or the MIN_CD, MAX_CD parameters will be requested.

F8: Outputs to a file or the printer a table which contains the valid number of pixels, mean, standard deviation and coefficient of determination for each input image file.

F9: Outputs to the screen a table which contains the valid number of pixels, mean, standard deviation and coefficient of determination for each input image file.

F10: Generates new blotch regions to be used for collecting statistics data. Once the key is pressed, a new function key set will be defined and the blotch areas created previously, if any, and the cursor will be displayed. At that time, the user can move the cursor around and press F1 to define a new vertex or F2 to erase the last vertex. As many as 500 vertices can be defined for each blotch area and up to 10 blotch areas can be defined. Key F3 is used to close the blotch area (connect the last vertex and the first vertex and fill the area with the color defined in the current graphics palette). Key F4 is used to erase a blotch area. This can only be done when the cursor is inside the blotch area. Key F5 is used to change the current graphics palette. Key ALT F9 allows the user to save current blotch areas into a file. Key ALT F10 allows the user to restore a blotch area from a file. After the blotch areas are created, the ESC key has to be used to quit this session. Once the new blotch areas are generated, all the input image files will be redisplayed one by one to get a new set of statistics data.

ALT F1: Toggles the function key menu display on/off.

ALT F5: Toggles the displayed image on and off.

ALT F6: Toggles the overlay graphics (blotched regions and the plot) on and off.

ALT F7: Toggles the blotched regions on and off.

ALT F8: Clears all the graphics which have the same color as defined in the current BPAL graphics palette.

ALT F9: Clears all overlay graphics.

MOUSE RIGHT BUTTON : Same as ALT F1.

PROGRAM NAME: VARIOG
DATE: 10/28/91
MENU: STAT2

DESCRIPTION: This program plots variograms, graphs of semivariance vs. lags, of the currently displayed image over a user defined line, rectangular box, or parallelogram. The calculations can be based on gray levels, pigment concentrations, or any linearly scaled data units. A user specified range may be used to exclude land, clouds, or other invalid pixel values during the calculations. The number of lags is determined by the number of pixels along the line, along the horizontal or vertical side of the box, or along the first or second side of the parallelogram. For all cases, the semivariance of a lag is obtained by summing the squares of the differences for valid pixel pairs divided by two times the total valid pairs (i.e., all the pixels are weighted equally). For more information, see Yoder et al. (1987).

PARAMETERS:

(1) **MODE** specifies the data type of the displayed image. A value of "1" (the default value) should be entered if the pixel values of the displayed image represent data (such as temperature) that are linearly related to gray levels. A value of "2" should be entered if they represent pigment concentrations (mg/m3).

(2) **FACTOR** is a non-negative scaling factor which is used only if MODE=1, i.e. the data-to-gray scale mapping function is linear for the displayed image. It is ignored when MODE=2. If FACTOR is positive, it will represent the factor by which to divide the gray values in order to convert them into actual data values. If zero is entered, the slope and intercept for this mapping function will be obtained from the header of the disk file for the displayed image. In order to retain the gray values, a "1" (the default value) should be entered ; for sea surface temperature (SST), enter 8; for water radiance data, enter 85; for aerosol radiance data, enter 100.

(3) **RANGE** defines the range of the pixel values to use for the calculations of semivariance. Two values should be entered that conform to the units of the displayed image (i.e. pigment concentration or units linearly proportional to gray levels) as specified by MODE and FACTOR. Pixel values less than the smaller RANGE value and those greater than the larger RANGE value will be excluded from the calculations. For example, to exclude land and cloud pixels for a level 2 CZCS image, the RANGE values should be 1.0 and 254.0 (the default values) for gray levels (MODE=1 and FACTOR=1) or 0.04093 and 45.0 for pigment concentrations (MODE=2).

DYNAMIC PARAMETERS:

I. Parameters for generating the plot.

(1) **DIR** indicates the direction for which to calculate the semivariance and is used for a box or parallelogram only. A "1" is entered to indicate that the calculations will be along the horizontal direction of a box or in the direction of the first defined side (i.e., between the first and second corners) of a parallelogram. A "2" is entered to indicate that the calculation will be along the vertical direction of a box or the second side of a parallelogram (between the second and third corners). This parameter refers to the most recently defined area. The initial default value is 1.

(2) **G_PAL** is the graphics palette to be used for the variogram.

(3) **XLABEL** is the label for the X axis of the variogram and may contain up to 40 characters. Upper and lower case letters and other characters may be used. The initial default label is "LAGS"; subsequently, the previously entered label is used as the default.

(4) **YLABEL** is the label for the Y axis of the variogram and may contain up to 40 characters. Upper and lower case letters and other characters may be used. The initial default label is "SEMIVARIANCE"; subsequently, the previously entered label is used as the default.

(5) **TITLE** is the title for the variogram. It may contain up to 40 characters and will appear below the graph. Upper and lower case letters and other characters may be used.

II. Parameters for outputting the plot data to a file.

(1) **O_FIL** is the output file name which will contain the lag numbers in the first column, the corresponding semivariance values in the second column, and the number of observations in the third column. A discrete character plot may also be generated, depending on the parameter PFLAG, after the third column. A name of "CON" for this parameter will send the output to the screen and "LPT1" or "LPT2" will send the output to the printer.

(2) **PFLAG** may have a value of "Y" or "N" to specify whether or not to generate a discrete character plot in the output file.

(3) **DIR** is analogous to DIR of DYNAMIC PARAMETERS I.

III. Parameters for clearing the graphics palette

(1) **CLR_PAL** is the number of the palette from which to clear overlay graphics. A "-1" is to used to clear all overlay graphics.

FUNCTION KEY DEFINITIONS:

ESC: Exits the program.

F1: Defines a line over which the semivariances of different lags are to be obtained. More than one line segment may be used in order to approximate a curved line.

F2: Defines a box (a rectangle with horizontal and vertical sides, i.e. sides along the pixel or line direction) over which the semivariances of different lags are to be obtained.

F3: Defines a parallelogram over which the semivariances of different lags are to be obtained. Three corners will need to be specified in a clockwise or counter-clockwise direction using the left and/or right mouse buttons.

F4: Asks the user to enter parameters (see DYNAMIC PARAMETERS I) for semivariance calculations on different lags and plots the variogram.

F5: Outputs the plot data to the screen, the printer, or an ASCII file. The parameters of DYNAMIC PARAMETERS II will be requested.

F6: Displays the next image frame buffer.

F7: Turns all graphics palettes on/off.

F8: Turns the displayed image on/off.

F9: Increases the current graphics palette by 1 or resets it to 1 if the value is greater than 7. The current graphics palette is used for defining the line (F1), box (F2), or parallelogram (F3).

F10: Clears all the overlay graphics or a specified graphics palette. The parameter CLR_PAL will be requested.

ALT F9: Displays the current cursor position.

ALT F10: Requests new values for parameters MODE, FACTOR and RANGE (see PARAMETERS)

ALT F1: Toggles function key menu display on/off.

MOUSE RIGHT BUTTON - Toggles function key menu display on/off.

PROGRAM NAME: WINDOW
DATE: 10/28/91
MENU: CZCSIN

DESCRIPTION: This program allows the user to extract subscenes of interest with a window (i.e. box) from a full size (1 x 1 resolution) CZCS scene. The size of the window can be changed interactively to define a subscene with different subsampling reduction factors. The box size is measured in full resolution image pixels, not monitor pixels. The input file should be a full size CZCS disk resident file generated by the program TP2DSK or DSK2DSK. The output file is 513 blocks in size with the first block being a header record followed by 512 data blocks (512x512 samples).

PARAMETERS:

(1) **INFILE** is the file name of the input full size CZCS scene. The file should be generated by the program TP2DSK or DSK2DSK.

(2) **FRMBUF** is the frame buffer for the full size image to be displayed. Only frame buffers 1 to 3 can be used.

(3) **PALETTE** indicates the graphics palette to be used for marking the window box. An integer number range from 1 to 7 should be entered.

DYNAMIC PARAMETERS:

I. Used for saving window image(s) into file(s).

(1) **OUT** is the output image filename. The exact file name will be used if a file for only one band of the image is to be created (key F1). If files for all bands are to be created (F2), they will be named by adding the corresponding digit from "1" to "6" at the end of the root name (prior to the period "." of any name extension). For example, if OUT="BAND.IMG", the output files generated will be "BAND1.IMG", "BAND2.IMG", ..., and "BAND6.IMG". When key F2 is used, the root name (excluding the extension) must be seven characters or less.

II. Used for moving the window box to a specified location.

(1) **LAT** is the latitude to where the upper left corner of the window box is to be moved on the current image. The value may be in decimal degrees, DMS format or radians (see help text for UNITS)

(2) **LON** is the longitude to where the upper left corner of the window box is to be moved on the current image. The value may be in decimal degrees, DMS format or radians (see help text for UNITS)

(3) **UNITS** is the units of LAT and LON :

1. Decimal degrees (initial default value).
2. DMS format, sDDDMMMSSS.SS, where s is for the sign, DDD is for degrees, MMM is for minutes and SSS.SS is for seconds of an arc (for example, -75030000.00 DMS is equal to -75.5 degrees, 163006000 is equal to 163.1 degrees).
3. Radians.

Note that modulo arithmetic is used for all three types of units. For example, -100.0, 260.0, 620.0, etc., are all equivalent degrees and may be entered for 100 west longitude.

FUNCTION KEY DEFINITIONS

ESC: Exits the program.

F1: Allows the user to save the section of the current image within the window to a disk file. The user will be prompted for the output file name. The output file will be a 512*512 subscene as defined by the windowed area. The pixel and line spacing will depend on the size of the window, e.g. a 1024x1024 box will generated a file subsampled by a factor of 2 in both pixel and line directions. A control file with the file extension .CTL will also be generated.

F2: Allows the user to save the section of the current image within the window, along with the corresponding part of its sister images, i.e. the other bands, to disk files. The user will be prompted for the output file name. The output file name is only used for the root. Six output files with names formed by appending indices of 1 to 6 to the root file will be generated. All the six output files will be 512x512 subscenes extracted from the corresponding input full size images on the same windowed area. A control file with the same root file name but with the file extension .CTL will also be generated.

F3: Changes the box size in the vertical direction. The box is initially 512 pixels in the vertical direction. When this button is depressed, it increases to 1024, when depressed again it goes to 1536, and finally to 3072 before returning to 512.

F4: Changes the box size in the horizontal direction. The box is initially 512 pixels in the horizontal direction. When this button is depressed, it increases to 1024, when depressed again it goes to 1536, and finally to 2048 before returning to 512. These pixel values are in full-resolution equivalents. This means that if the original image had 2048 pixels, the initial 512 box will span only one-fourth the width of the scene or 128 monitor pixels.

F5: Marks the current window box with the color defined in output look-up table palette parameter PALETTE.

F6: Displays the coordinates of the window box. The pixel/line and latitude/longitude coordinates of the upper left and lower right corners of the box are displayed.

F7: Allows the user to specify a latitude and longitude at which to place the upper left corner of the window box.

ALT F1: Toggles the function key menu display on/off.

MOUSE RIGHT BUTTON - Same as ALT F1.

PROGRAM NAME: WTKLM
DATE: 10/28/91
MENU: L2PROD

DESCRIPTION: The program WTKLM is another version of the program L2MULT. It uses the Weitek numerical coprocessor and runs under the protected mode with the Phar Lap DOS-Extender. The program accepts the input parameters entered by the user and writes them out to a temporary file with the name "ZZHHMMSS.TMP", where the HH, MM and SS are current hour, minute and second of the system clock. Then, it invokes the protected mode program WTKL2MLT.EXE (bound with DOS-Extender) and passes in the temporary parameter file name. The program WTKL2MLT.EXE reads in the input parameters from the temporary parameter file and then generates the level-2 products. The temporary parameter file will be deleted after WTKL2MLT.EXE is executed. Actually, WTKLM is just a driver program (run under real mode) and the main process program is the WTKL2MLT.EXE (run under protected mode.) The user will notice some delay when WTKLM invokes the program WTKL2MLT.EXE.

PARAMETERS:
All input parameters used by this program are the same as those of the program L2MULT.

PROGRAM NAME: WTKMP
DATE: 10/28/91
MENU: PROJECTN

DESCRIPTION: The program WTKMP is another version of the program MAPIMG. It uses the Weitek numerical coprocessor and runs under the protected mode with the Phar Lap DOS-Extender. The program accepts the input parameters entered by the user and writes them out to a temporary file with the name "ZZHHMMSS.TMP", where the HH, MM and SS are current hour, minute and second of the system clock. Then, it invokes the protected mode program WTKMPIMG.EXE (bound with DOS-Extender) and passes in the temporary parameter file name. WTKMPIMG.EXE reads the input parameters from the temporary file and generates the mapped image. The temporary parameter file will be deleted after WTKMPIMG.EXE is executed. Actually, WTKMP is just a driver program (run under real mode) and the main process program is WTKMPIMG.EXE (run under protected mode). The user will notice some delay when WTKMP invokes the program WTKMPIMG.EXE.

PARAMETERS:
All the input parameters for this program are the same as those of the program MAPIMG.

PROGRAM NAME: XCORR
DATE: 10/28/91
MENU: STAT2

DESCRIPTION: This program plots cross correlation vs. lags for two images over a user defined line, rectangular box, or parallelogram. The calculations can be based on gray levels, pigment concentrations, or any linearly scaled data units. A user specified range can be used to exclude land, clouds, or other invalid pixel values during the calculations. For a box or a parallelogram, the cross correlation for each lag is calculated line by line then averaged across all the lines. The number of lags calculated are about half of the total pixels on the negative and positive sides of the line, the horizontal or vertical side of the box, or the first or second side of the parallelogram. The cross correlation of the positive and negative lags are calculated by moving the second image data array to the left and right of the first image data array separately.

PARAMETERS:

(1) **FBUF1** specifies the image frame buffer of the first of two images undergoing the cross correlation analysis.

(2) **FBUF2** specifies the image frame buffer of the second of two images undergoing the cross correlation analysis.

(3) **MODE1** defines whether the image in FBUF1 is scaled linearly or is in pigment concentration. A value of "1" (the default value) should be entered if the pixel values of the FBUF1 image represent data (such as temperature) that are linearly related to gray levels. A value of "2" should be entered if they represent pigment concentrations (mg/m3).

(4) **MODE2** is similar to MODE1 but applies to the image in FBUF2.

(5) **FACTOR1** is a non-negative scaling factor which is used only if MODE1=1, i.e. the data-to-gray scale mapping function is linear for the FBUF1 image. It is ignored when MODE1=2. If FACTOR1 is positive, it represents the factor by which to divide the gray values of FBUF1 pixels in order to convert them into actual data values. If zero is entered, the slope and intercept for this mapping function will be obtained from the header of the disk file for the FBUF1 image. In order to retain the gray values, a "1" (the default value) should be entered ; for sea surface temperature (SST), enter 8; for water radiance data, enter 85; for aerosol radiance data, enter 100.

(6) **FACTOR2** is the linear, data-to-gray scale mapping function for FBUF2. Comments analogous to those of FACTOR1 apply here.

(7) **RANGE1** defines the range of FBUF1 pixel values to use for the calculations of the cross correlations. Two values should be entered for this parameter. These values should conform to the units of the FBUF1 image (i.e. pigment concentration or units linearly proportional to gray levels) as specified by MODE1 and FACTOR1. Pixel values less than the smaller RANGE1

value and those greater than the larger RANGE1 value will be excluded from the calculations. For example, to exclude land and cloud pixels for a level-2 CZCS image, the RANGE1 values should be 1.0 and 254.0 (the default values) for gray levels (MODE1=1 and FACTOR1=1) or 0.04093 and 45.0 for pigment concentrations (MODE1=2).

(8) **RANGE2** defines the range for valid FBUF2 values in data units. Comments analogous to those of RANGE1 apply here.

DYNAMIC PARAMETERS:

I. Parameters for generating the plot.

(1) **DIR** indicates the direction for which to calculate the cross correlations and is used for a box or parallelogram only. A "1" is entered to indicate that the calculations will be along the horizontal direction of a box or in the direction of the first defined side (i.e., between the first and second corners) of a parallelogram. A "2" is entered to indicate that the calculation will be along the vertical direction of a box or the second side of a parallelogram (between the second and third corners). This parameter refers to the most recently defined area. The initial default value is 1.

(2) **G_PAL** is the graphics palette to be used for the plot of the cross correlation vs. lags.

(3) **XLABEL** is the label for the X axis of the cross correlation vs. lags plot and may contain up to 40 characters. Upper and lower case letters and other characters may be used. The initial default label is "LAGS"; subsequently, the previously entered label is used as the default.

(4) **YLABEL** is the label for the Y axis of the cross correlation vs. lags plot and may contain up to 40 characters. Upper and lower case letters and other characters may be used. The initial default label is "CROSS CORRELATION"; subsequently, the previously entered label is used as the default.

(5) **TITLE** is the title for the plot of the cross correlation vs. lags. It may contain up to 40 characters and will appear below the plot. Upper and lower case letters and other characters may be used.

II. Parameters for outputting the plot data to a file.

(1) **O_FIL** is the output file name which will contain the lag numbers in the first column, the corresponding cross correlation values in the second column, and the number of observations in the third column. A discrete character plot may also be generated, depending on the parameter PFLAG, after the third column. A name of "CON" for this parameter will send the output to the screen and "LPT1" or "LPT2" will send the output to the printer.

(2) **PFLAG** may have a value of "Y" or "N" to specify whether or not to generate a discrete character plot in the output file.

(3) **DIR** is analogous to DIR of DYNAMIC PARAMETERS I.

III. Parameters for clearing the graphics palette

(1) **CLR_PAL** is the number of the palette from which to clear overlay graphics. A "-1" is to used to clear all overlay graphics.

FUNCTION KEY DEFINITIONS:

ESC: Exits the program.

F1: Defines a line over which the cross correlations of different lags are to be obtained. More than one line segment may be used in order to approximate a curved line.

F2: Defines a box (a rectangle with horizontal and vertical sides, i.e. sides along the pixel or line direction) over which the cross correlations of different lags are to be obtained.

F3: Defines a parallelogram over which the cross correlations of different lags are to be obtained. Three corners will need to be specified in a clockwise or counter-clockwise direction using the left and/or right mouse buttons.

F4: Asks the user to enter parameters (see DYNAMIC PARAMETERS I) for cross correlation calculations on different lags and plots the results.

F5: Outputs the plot data to the screen, the printer, or an ASCII file. The parameters of DYNAMIC PARAMETERS II will be requested.

F6: Switches the displayed image between FBUF1 and FBUF2.

F7: Turns all graphics palettes on/off.

F8: Turns the displayed image on/off.

F9: Increases the current graphics palette by 1 or resets it to 1 if the value is greater than 7. The current graphics palette is used for defining the line (F1), box (F2), or parallelogram (F3).

F10: Clears all the overlay graphics or a specified graphics palette. The parameter CLR_PAL will be requested.

ALT F9: Displays the current cursor position.

ALT F10: Requests new values for parameters MODE, FACTOR and RANGE (see PARAMETERS)

ALT F1: Toggles function key menu display on/off.

MOUSE RIGHT BUTTON - Toggles function key menu display on/off.

PROGRAM NAME: ZOOM
DATE: 10/28/91
MENU: FRMBUF

DESCRIPTION: This program will magnify 2, 4, or 8 times the image currently displayed on the MVP-AT. While the image is zoomed, one may roam around it (i.e., shift it) by moving the mouse. The portion zoomed is always the upper left corner of the image currently displayed. If any overlay graphics are present, they will also be zoomed and shifted to the same extent. Note that while roaming, the image will wrap around according to the shift. Also, the minimum shift in the X direction is four or eight pixels depending on whether the MVP-AT has been setup for interlaced mode (four pixels) or non-interlaced mode (eight pixels).

PARAMETERS:
There are no parameters.

DYNAMIC PARAMETERS:

I. Used in saving the zoomed and panned image and/or graphics into disk file(s).

(1) **O_FIL** is the name of the file to create for saving the currently zoomed and panned image. If the name is blank, the image will not be saved. A blank 512-byte header block will be added to the beginning of the file. Note that, if there is a wrap-around on the displayed image, only the upper left part will be saved.

(2) **G_FIL** is the name of the file to create for saving the currently zoomed and panned graphics. If the name is blank, the graphics will not be saved. A header block is not added to the graphics file. Note that, if there is a wrap-around on the displayed graphics, only the upper left part will be saved.

FUNCTION KEY DEFINITIONS:

ESC: Exits the program.

F1: Toggles the image being displayed between the original image and the zoomed image. A box will also be drawn around the area presently being magnified when the original image is displayed.

F2: Increases the zoom factor by a factor of two up to eight.

F3: Decreases the zoom factor by a factor of two down to one.

F4: Turns the cursor on and off.

F5: Allows the user to save the currently zoomed and panned image and/or the overlay graphics into files.

REFERENCES

Arnone, R.A., and P.E. LaViolette, A method of selecting optimal Angstrom coefficients to obtain quantitative ocean color data from Nimbus-7 CZCS, Soc. Photo-Opt. Inst. Eng., 489, Ocean Optics VII, 187-194, 1984.

Austin, R.W., and T.J. Petzold, The determination of the diffuse attenuation coefficient of sea water using the Coastal Zone Color Scanner, Oceanography from Space, J.F.R. Gower (Ed.), Plenum Press, New York, pp. 239-256, 1981.

Barale, V., C.R. McClain, and P. Malanotte-Rizzoli, Space and time variability of the surface color field in the northern Adriatic Sea, J. Geophys. Res., 91(C11), 12,957-12,974, 1986.

Barbieri, R.W., C.R. McClain, and D. L. Endres, Methodology for interpretation of SST retrievals using the AVHRR split window algorithm, NASA Tech. Mem. 85100, 1983.

Bernstein, R.L., Sea surface temperature estimation using the NOAA-6 satellite Advanced Very High Resolution Radiometer, J. Geophys. Res., 87(C12), 9,455-9,465, 1982.

Brown, O.B., J.W. Brown, and R.H. Evans, Calibration of Advanced Very High Resolution Radiometer infrared observations, J. Geophys. Res., 90(C6), 11,667-11,677, 1985.

Clark, D.K., Phytoplankton algorithms for the Nimbus-7 CZCS, Oceanography from Space, J.R.F Gower, Ed., Plenum Press, New York, 1981.

Darzi, M., J. Chen, J.K. Firestone, and C.R. McClain, SEAPAK: A satellite image analysis system for oceanographic research, Preprint Volume, 5th Intl. Conf. on Interactive and Information Processing Systems for Meteorology, Oceanography, and Hydrology, Amer. Meteorological Soc., Boston, pp. 26-32, 1989.

Darzi, M., J.K. Firestone, G. Fu, E.-n. Yeh, and C.R. McClain, Current efforts regarding the SEAPAK oceanographic analysis software system, Preprint Volume, 7th Intl. Conf. on Interactive and Information Processing Systems for Meteorology, Oceanography, and Hydrology, Amer. Meteorological Soc., Boston, pp. 109-115, 1991.

Esaias, W.E., G.C. Feldman, C.R. McClain, and J.A. Elrod, Monthly satellite-derived phytoplankton pigment distribution for the North Atlantic Ocean basin, EOS, 67(44), 835-837, 1986.

Feldman, C., N. Kuring, C. Ng, W. Esaias, C. McClain, J. Elrod, N. Maynard, D. Endres, R. Evans, J. Brown, S. Walsh, M. Carle, and G. Podesta, Ocean color: Availability of the global data set, EOS, 70(23), 634-641, 1989.

Firestone, J.K., and J.F. Chen, Low-Cost Image Processing for the Laboratory for Oceans: A Feasibility Study, GSC-TR8748, General Sciences Corporation, Laurel, MD, 1987.

Firestone, J.K., G. Fu, J. Chen, M. Darzi and C.R. McClain, PC-SEAPAK: A state-of-the-art image display and analysis system for NASA's oceanographic research program, Preprint Volume, 5th Intl. Conf. on Interactive and Information Processing Systems for Meteorology, Oceanography, and Hydrology, Amer. Meteorological Soc., Boston, pp. 33-40, 1989.

Firestone, J.K., G. Fu, M. Darzi and C.R. McClain, NASA's SEAPAK software for oceanographic data analysis, Preprint Volume, 6th Intl. Conf. on Interactive and Information Processing Systems for Meteorology, Oceanography, and Hydrology, Amer. Meteorological Soc., Boston, pp. 260-267, 1990.

Firestone, J.K., M. Darzi, G. Fu, E.-n. Yeh, and C.R. McClain, Oceanographic data analysis with NASA's SEAPAK software, Proceedings, ACSM/ASPRS/Auto Carto 10 Conf., Am. Soc. for Photogrammetry and Remote Sensing, Falls Church, VA, pp. 145-161, 1991.

Fu, G., M. Darzi, J.K. Firestone, and C.R. McClain, SEAPAK: A comprehensive oceanographic analysis software system, Science and Technology Conference Proceedings, Marine Technology Soc., Washington, DC, pp. 527-532, 1990.

Gordon, H.R., Reduction of error introduced in the processing of color zone color scanner-type imagery resulting from sensor calibration and solar irradiance uncertainty, Appl. Opt., 20, 207-210, 1981.

Gordon, H.R., Visible calibration of ocean-viewing sensors, Remote Sens. of Environ., 22, 103-126, 1987.

Gordon, H.R., J.W. Brown, and R.H. Evans, Exact Rayleigh scattering calculations for use with the Nimbus-7 Coastal Zone Color Scanner, Appl. Opt., 27(5), 862-871, 1988.

Gordon, H.R., and D.K. Clark, Clear water radiances for atmospheric correction of Coastal Zone Color Scanner imagery, Appl. Opt., 20(24), 4,175-4,180, 1981.

Gordon, H.R., D.K. Clark, J.W. Brown, O.B. Brown, R.H. Evans, and W.W. Broenkow, Phytoplankton pigment concentrations in the Middle Atlantic Bight: Comparison of ship determinations and CZCS estimates, Appl. Opt., 22(1), 20-36, 1983.

Gordon, H.R., J.W. Brown, O.B. Brown, R.H. Evans, and D.K. Clark, Nimbus-7 CZCS: Reduction of its radiometric sensitivity with time, Appl. Optics, 22(24), 3,929-3,931, 1983a.

Hovis, W.A., J.S. Knoll, and G.R. Smith, Aircraft measurements for calibration of an orbiting spacecraft center, Appl. Opt., 24, 407-410, 1985.

Kaufman, Y.J., and B.N. Holben, Calibration of the AVHRR visible and near-IR bands by atmospheric scattering, ocean glint and desert reflection, J. Applied Meterol., in press, 1991.

Kidwell, K.B., (Ed.), NOAA Polar Orbiter Data Users Guide, rev. Dec. 1988, NOAA/NESDIS/NCDC/SDSD, U.S. Dept. of Commerce, 1988.

McClain, C.R., J. Chen, M. Darzi, J. Firestone, and D. Endres, SEAPAK User's Guide, Ver. 1.0, NASA Tech. Mem. 100728, 1989.

McClain, C.R., M. Darzi, J. Firestone, E.-n. Yeh, and D. Endres, SEAPAK User's Guide, Ver. 2.0, Vol. I--System Description, 158 pp., Vol. II--Descriptions of Programs, 586 pp., NASA Tech. Mem. 100728, 1991.

McClain, C.R., J.A. Yoder, L.P. Atkinson, J.O. Blanton, T.N. Lee, J.J. Singer, and F. Muller-Karger, Variability of surface pigment concentrations in the South Atlantic Bight, J. Geophys. Res., 93(C9), 10,675-10,697, 1988

McClain, E.P., Split-window and triple-window sea surface temperature determinations from satellite measurements, Proc. ICES Statutory Meeting, Woods Hole, MA, 6-10 October, 1981.

McClain, E.P., W.G. Pichel, and C.C. Walton, Comparative performance of AVHRR-based multichannel sea surface temperatures, J. Geophys. Res., 90(C6), 11,587-11,601, 1985.

Mueller, J.L., Nimbus-7 CZCS: Confirmation of its radiometric sensitivity decay rate through 1982, Appl. Opt., 24, 1043-1047, 1985.

Mueller, J.L., Nimbus-7 CZCS: Electronic overshoot due to cloud reflectance, Appl. Opt., 27(3), 438-440, 1988.

Muller-Karger, F.E., C.R. McClain, R.N. Sambrotto, and G.C. Ray, A comparison of ship and CZCS-mapped distributions of phytoplankton in the southeastern Bering Sea, J. Geophys. Res., 95(C7), 11,483-11,499, 1990.

Murray, C. W., J. L. Mueller, and H. J. Zwally, Matrix Partioning and EOF/Principal Component Analysis of Antarctic Sea Ice Brightness Temperatures, NASA TM 83916, April 1984.

Planet, W.G. (Ed.), Data Extraction and Calibration of Tiros-N/NOAA Radiometers, NOAA Tech. Mem. NESS 107, Rev. 1, U.S. Dept. of Commerce, Washington, DC, 1988.

Press, W.H., B.P. Flannery, S.A. Teukolsky, and W.T. Vetterling, Numerical Recipes: The Art of Scientific Computing, Cambridge Univ. Press, Cambridge, U.K., 1986.

Smith, R.C., and W.H. Wilson, Ship and satellite bio-optical research in the California Bight, Oceanography from Space, J.F.R. Gower (Ed.), Plenum Press, New York, pp. 281-294, 1981.

Snyder, J.P., Map Projections Used by the U.S. Geological Survey, 2nd ed., Bulletin 1532, U.S. Geological Survey, 1982.

Strong, A.E., and E.P. McClain, Improved ocean surface temperatures from space: Comparisons with drifting buoys, Bull. AMS, 65(2), 138-142, 1984.

Viollier, M., Radiance calibration of the Coastal Zone Color Scanner: A proposed adjustment, Appl. Opt., 21, 1142-1145, 1982.

Weinreb, M.P., G. Hamilton, and S. Brown, Nonlinearity corrections in calibration of Advanced Very High Resolution Radiometer infrared channels, J. Geophys. Res., 95(C5), 7,381-7,388, 1990.

Williams, S.P., E.F. Szajna, and W.A. Hovis, NIMBUS 7 Coastal Zone Color Scanner (CZCS): Level 2 Data Product User's Guide, NASA Tech. Mem. 86202, 1985a.

Williams, S.P., E.F. Szajna, and W.A. Hovis, NIMBUS 7 Coastal Zone Color Scanner (CZCS): Level 1 Data Product User's Guide, NASA Tech. Mem. 86203, 1985b.

Yoder, J.A., C.R. McClain, J.O. Blanton and L.-Y. Oey, Spatial scales in CZCS-chlorophyll imagery of the southeastern U.S. continental shelf, Limn. Oceanogr., 32(4), 929-941, 1987.

GLOSSARY

ASCII Acronym for American Standard Code for Information Interchange, a code for representing an alphanumeric and symbol character set in binary.

AVHRR Advanced Very High Resolution Radiometer. This is one of the sensors aboard the NOAA series of satellites. There are four channels for even numbered satellites such as NOAA-10 and five channels for odd numbered satellites such as NOAA-9. The radiance bands measured are 0.58-0.68, 0.725-1.1, 3.55-3.93, 10.5-11.5, and, for odd-numbered satellites, 11.5-12.5 um. (See **GAC** and **LAC**.)

Band A wavelength range of spectrum within which a sensor makes measurements. (See **AVHRR**, **CZCS**, **Channel**.)

Block A quantity of storage or data in bytes. Equals 512 bytes unless otherwise specified.

Blotch A colored-in area on a overlay graphics which normally corresponds to a region of interest for associated image(s). A blotch enables the system to differentiate between the areas inside and outside this region when performing analyses on images. (See **Region of Interest**.)

Case 1 Water Areas where phytoplankton and derivatives are the primary determinant of the water's optical characteristics. Such areas are normally in the open ocean.

Case 2 Water Areas where water sediments are the primary determinant of the water's optical characteristics. Such areas are normally coastal regions.

Channel Spectral band at which measurements are made by a radiometer. The CZCS has six such channels; the AVHRR, four or five. (See **Band**.)

Counts Refers to the digitized value of the radiance measured by a radiometer for an individual point (pixel). Usually applies to data prior to various corrections which convert them to physical units. CZCS uses 8 bits per pixel so count values range from 0 to 255. Count values are used to generate level-1 images.

CRT CZCS calibrated radiance (level-1) tape.

CZCS Coastal Zone Color Scanner, a scanning radiometer aboard the Nimbus-7 satellite with channels at 0.433-0.453, 0.510-0.530, 0.540-0.560, 0.660-0.680, 0.700-0.800, and 10.5-12.5 um, and a resolution at nadir of 825 m^2. It was functional from November 1978 to June 1986.

Display Coordinates Refers to the vertical and horizontal coordinates of a picture element on an image display. Also called **TV coordinates.**

DOS The acronym for Disk Operating System, the operating system for which PC-SEAPAK was developed.

DOS extender Provides for a 80386 chip a protected mode run-time environment for application programs running undr DOS. Allows memory greater than 640 Kbytes to be accessed by application programs. (See **DOS, real mode, protected mode.**)

Dropping an Image Refers to the loading of a digital image contained in a disk file into a frame buffer of the MVP-AT.

DSP An image analysis system developed for CZCS and AVHRR data at the University of Miami.

Frame Buffer Random access memory area in the MVP-AT used to store digitized images. For PC-SEAPAK, the MVP-AT's on-board one megabyte of memory is configured into four 512x512x8-bit frame buffers. These four buffers are numbered 0 to 3. Some PC-SEAPAK programs permit the memory to be accessed as two 512x512x16-bit frame buffers, numbered 4 and 5. In that case, the 16-bit buffer 4 is equivalent to the two 8-bit buffers 0 and 1 and buffer 5 is equivalent to buffers 2 and 3. (See **MVP-AT.**)

Full-Scan Image A CZCS image whose width is 1,968 pixels--a full scan line. The number of lines may vary. The image is a full-resolution image produced by the program TP2DSK. The program WINDOW may be used to display such images and to create regular, 512x512 PC-SEAPAK images. Also referred to as full-width or full-size images.

GAC Global Area Coverage, the lower resolution coverage provided by the AVHRR data. The resolution is approximately 4 km.

Gray Level — Digitized value used to represent a pixel quantity. When displayed on an image system, such a value is assigned a gray shade or level. Normally, 8 bits per pixel are used for this digitization so that gray levels range from 0 (black) to 255 (white). The quantity represented may be count, level-1, level-2, or level-3 data.

GSFC — NASA's Goddard Space Flight Center in Greenbelt, Maryland.

HRPT — High resolution picture transmission, direct readout AVHRR data. (See **LAC**.)

Image File — A 512x512-byte, flat file containing gray level values for an image which can be displayed on the MVP-AT. Each byte represents one image pixel. Additional 512-byte records containing the header information may precede the image data. PC-SEAPAK images typically contain one such header record.

I/O — The acronym for input/output.

ILUT — Input look-up table set. It modifies the input gray-scale values before they enter the frame buffer. For the MVP-AT, each of the red, green and blue look-up tables in the ILUT is divided into 32 256-byte palettes numbered 0 to 31.

LAC — Local Area Coverage, the higher resolution coverage provided by the AVHRR. The resolution is approximately 1 km.

Lag — A space or time interval separating points in correlation and spectral analyses.

Land Mask — A graphics depicting the location of land; often used as an overlay to satellite imagery. May be used as a blotch to exclude processing of land or water areas in an image.

Level 1 — Refers to a satellite image whose data have been corrected for sensor calibration. For CZCS, these data represent observed radiance values derived from the 8-bit count transmitted by the satellite.

Level 2 Refers to a satellite image whose data represents a geophysical parameter derived from level-1 images. To derive level-2 CZCS images, level-1 images from various bands of the same scene are used. Level-2 images include images such as those of water radiances, pigment concentrations, aerosol radiances, and diffuse attenuation for the CZCS or SST images for the AVHRR.

Level 3 Refers to a satellite image which has been mapped to a non-satellite perspective projection. Such mapping is normally (but not necessarily) performed on level-2 images.

Line Refers to the vertical location or coordinate of a picture element when used in conjunction with "pixel." (See **Display Coordinates**.)

LUT Look-up table. In MVP-AT, there is an input look-up table set (**ILUT**) and an output look-up table set (**OLUT**).

MS-DOS Acronym for Microsoft Disk Operating System. (See **DOS**.)

MVP-AT Image processing board, used by PC-SEAPAK, for the IBM PC AT from Matrox Electronic Systems Limited. (See **Frame Buffer**.)

NASA National Aeronautics and Space Administration.

NOAA National Oceanic and Atmospheric Administration, Dept. of Commerce.

OLUT Output look-up table set. It modifies the gray-scale intensity fed from the frame buffer to the output display device. For the MVP-AT, each of the red, green and blue look-up table in the OLUT is divided into 32 256-byte palettes numbered 0 to 31.

Overlay Refers to graphics that are displayed non-destructively in conjunction with an image (i.e., without altering the image). In PC-SEAPAK, frame buffer 0 is normally used to display overlays. An overlay may be saved in, or restored to the frame buffer from, a graphics (or "blotch") file identical in structure to image files but without header blocks.

Palette The palette is a look-up table which contains red, green, and blue 256-byte look-up tables. For the MVP-AT, there are 32 palettes each for the input look-up table (**ILUT**) and for the output look-up table (**OLUT**).

Pixel 1. The smallest element of a digital image for which a value is assigned. This may refer to the individual points at which a radiometer takes measurements or the picture elements of an image display. (See **Sample**.)
2. Refers to the horizontal location or coordinate of a picture element when used in conjunction with "line." (See **Display Coordinates**.)

Protected mode Aspecial operating mode for 80286 and 80386 chips that supports memory protection, addressing using segment selectors, and 16bit or 32-bit (for 80386) instruction sets. (See **DOS**, **DOS extender**, **real mode**.)

Real mode The DOS operation mode for 8086/8088, 80286, and 80386 chips that allows up to one megabyte of memory addressed and does not provide any memory protection features. (See **DOS**, **DOS extender**, **protected mode**.)

Region of Interest (ROI) A portion of an image, corresponding to one or more blotch areas, to be processed by a program. Certain PC-SEAPAK programs allow either the blotch (inside) or non-blotch (outside) areas to be specified as the ROI. (See **Blotch**.)

Registration Refers to the process of manipulating an image so that its earth-surface points may be superimposed on those of another image or graphics display.

Remap To project an image using the PC-SEAPAK program MAPIMG to a standard map projection.

Sample Usually refers to a data point along a scan line but also used interchangeably with **pixel**.

Scan Line A line of measurements (**samples**) taken during one rotation of a satellite sensor across its flight path.

SCSI Small computer system interface, an industry standard interface that provides high-speed access to peripheral devices.

SST Sea surface temperature. In PC-SEAPAK, refers to level-2 data derived from the AVHRR.

TOMS The Total Ozone Mapping Spectrometer aboard the Nimbus-7 satellite. Uses backscattered ultraviolet to make total ozone measurements at a resolution of 50 to 150 km.

TV Coordinates Same as **display coordinates.**

APPENDIX
PC-SEAPAK HARDWARE AND SOFTWARE REQUIREMENTS AND OPTIONS

Lists of the hardware and software used for the PC-SEAPAK system are listed. Each item is preceded by a symbol to indicate whether it is required (*), recommended (!), or optional (o). Since the prices listed here are only meant as guidelines based on our most recent purchases or price lists we obtained from various vendors, they may not represent the lowest or most recent quotes available.

Basic System

Although PC-SEAPAK was developed on a COMPAQ Deskpro 386/20, it can be run on most 386-compatible computers with an AT bus, an EGA controller, and an Intel 80387 coprocessor. The COMPAQ Deskpro 386/20 has been discontinued and replaced by the Deskpro 386/25 and 386/33. PC-SEAPAK has been installed and run on the Deskpro 386/25 and 386/33 here at NASA/GSFC Oceans Computing Facility as well as by several user groups. The Deskpro 386/33 is recommended because of its better CPU speed and hardware configuration. **Note that conflict problems have been reported for the ZENITH computer using the MVP-AT image board and it is therefore not recommended.**

! Compaq Deskpro 386/33. Price: 5,165

4 MB memory
120 MB fixed disk drive with integrated controller
Integrated VGA controller
Eight expansion slots
1.44 MB 3.5" floppy drive
Enhanced keyboard

Vendor: Any Compaq computer dealer.

Comments:
1. Higher price for larger hard disk. 320 MB or 650 MB internal drives are available. If no other external hard disk or rewritable/erasable optical drive is added to the system, the 320 or 650 MB hard disk is recommended. One of the expansion slots in this system is occupied by the floppy diskette control card and is not free to use.
2. All compatible 386 computers with AT bus are also recommended.

! 80387 (33 MHz) floating-point coprocessor. Price: 300
Vendor: Any Compaq computer dealer.
Comments: 1. 33 MHz is necessary for COMPAQ 386/33.
2. Most of the PC-SEAPAK application programs are built with Microsoft's Fortran and C compilers and the Intel 80387 coprocessor to be used for numerical calculations. MicroWay's NDP Fortran-386 compiler is also used to build some large application programs to run under protected mode with the 80387 coprocessor or the Weitek 3167 chips.

o 2 MB 32-bit memory module. Price: 300
Vendor: Any Compaq computer dealer.
Comments: 1. At least 4 MB of memory is recommended.
2. More memory can be used for virtual disk or running multitasks under DESQview.
3. With Phar Lap's DOS-Extender and MicroWay's NDP Fortran-386 compiler, application programs as large as the memory limit can be built.

! Additional 3.5" 1.44 MB or 5.25" 1.2 MB floppy drive. Price: 182
Vendor: Any Compaq computer dealer.

! 3167 (Weitek) 33 MHz numerical coprocessor board. Price: 1,100
Vendors: 1. Any Compaq computer dealer.
2. MicroWay Inc.
Comments: 1. 33 MHz is necessary for the COMPAQ 386/33.
2. The Microsoft compilers (Fortran, C) cannot generate code for the Weitek chips. MicroWay's NDP Fortran-386 compiler is used to generate code for the Weitek chips.

o AST-VGA adapter. Price: 350
Vendor: AST Research, Inc.
Comments: 1. This is recommended only for the Deskpro 386/25. The Deskpro 386/33 already contains an integrated VGA adapter in the system board.
2. It also supports EGA and CGA which are necessary for PC-SEAPAK.

* Matrox MVP-AT image board. Price: 5,495
Vendors: 1. Any Matrox dealer.
2. Matrox Electronic Systems.
Comments: 1. The cable to connect the board and the monitor must be bought separately.
2. Two cable lengths are available and are listed below.

* 4' PC-OCABLE-4 video output cable. Price: 65
Vendors: 1. Any Matrox dealer.
2. Matrox Electronic Systems.
Comments: 1. This cable is used to connect the MVP-AT board and the image display monitor.
2. The 10' PC-OCABLE-10 is another option.

o 10' PC-ICABLE-10 video input cable. Price: 100
Vendors: 1. Any Matrox dealer.
2. Matrox Electronic Systems.
Comments: 1. This cable is used to connect the video input device and the MVP-AT board.
2. This is not necessary for PC-SEAPAK applications.

Monitors

o Compaq VGA monitor. Price: 496
Vendor: Any Compaq computer dealer.
Comments: 1. For regular text and graphic display.
2. Size, 14"; resolution, 640x480.

! NEC Multisync 3D or 4D. Price: 665
Vendor: Any microcomputer dealer.
Comments: 1. For regular text and graphic display.
2. Size, 14"; resolution. 1024x768.

! Mitsubishi HA3905L9 ADK. Price: 2,000
Vendor: Mitsubishi Electronics America, Inc.
Comments: 1. Size, 20"; resolution, 1024x800.
2. For interlace and non-interlace mode image display.

External Storage

! Cipher M990 tape drive, controller card, utility package and software library. Price: 7,090
Vendor: 1. Overland Data, Inc. (ODI)
2. Flagstaff Engineering, Inc.
Comments: 1. This is a 1/2" 9-track tape drive.
2. Cipher has a new model M995 tape drive to replace the old model M990.
3. Flagstaff Engineering, Inc. had sold the 9-track tape drive business to Overland Data, Inc.
4. Overland Data, Inc. sells its own controller card and software utility and library. ODI may carry over the Flagstaff Engineering, Inc.'s controller card and software for some time.

5. Only the Cipher M990 or M995 tape drive with the controller card and the software driver from Flagstaff Engineering, Inc. is supported in the PC-SEAPAK tape ingest programs. The ODI's controller card is not supported by the PC-SEAPAK tape ingest programs.
6. PC-SEAPAK supports disk ingest programs that allow the user to ingest data from disk if the data on the tape can be copied to the disk.
7. ODI has tape utility programs for its own controller that allow the user to copy the data from the tape to the disk.

! Storage and backup system:
Various DAT, 8mm, erasable optical disk systems are available.

Hard Copy

o HP LaserJet II. Price: 1,610
o Parallel interface cable. Price: 15
Vendor: Any Hewlett Packard dealer.
Comment: For high quality text output.

! HP PaintJet color printer. Price: 930
! Parallel interface cable. Price: 15
o Black pen cartridge. Price: 21
o Color pen cartridge. Price: 26
Vendor: Any Hewlett Packard dealer.
Comment: This printer will be supported as a hard copy device for images from PC-SEAPAK.

o Tektronix 4693RGB color screen printer. Price: 6,500
Vendor: Tektronix, Inc.
Comment: This can only handle non-interlaced inline RGB video signal.

o QMS ColorScript 100 Model 10 Price: 8,995
Vendor: QMS, Inc.
Comment: This can only handle postscript formatted file.

VAX-to-PC Communication

o 3Com thin-wire Ethernet controller with T-connector (model Etherlink II, 3C503). Price: 289
Vendor: 3Com Corporation.

Software

! DESQview 386. Price: 180
! DESQview 2.3. Price: 130
! QEMM-386 5.1. Price: 60

Vendors: 1. Any computer dealer.
2. Quarterdeck Office System.

Comment: 1. DESQview 386 is the combination of DESQview and QEMM-386.
2. DESQview is a multitasking DOS operating environment.
3. QEMM is a memory management program that allows the user to load memory resident programs to the high memory and to emulate the expanded memory by using the extended memory.

! Compaq MS-DOS 3.31 to 5.0. Price: 90

Vendor: Any Compaq computer dealer.

Comment: 1. The PC-SEAPAK is developed under Compaq MS-DOS 3.31.
2. PC-SEAPAK had been tested running under DOS 4.1 and 5.0.

o Microsoft Fortran compiler. Price: 280
o Microsoft C compiler. Price: 280
* Microsoft mouse (serial version) Price: 120

Vendor: Any microcomputer dealer.

o NDP Fortran 386 compiler for DOS. Price: 495

Vendor: MicroWay, Inc.

Comments: 1. This compiler generates native 80386 32-bit object code that runs in protected mode under MS-DOS.
2. It can also generate code to use Weitek chip.
3. To use with Phar Lap's 386 development system, it can generate application programs which use the whole memory available on the system.

o Phar Lap's 386 development system. Price: 525

Vendor: Phar Lap Software, Inc.

Comments: 1. This development system includes the DOS-Extender and the 386|ASM/LINK.
2. This package is necessary to link and run all application programs developed under NDP Fortran-386 compiler.

o Imager-AT interpreter command and software library. Price: 1,000

Vendors: 1. Any Matrox dealer.
2. Matrox Electronic Systems.

- o WordPerfect Version 5.1. Price: 180
  Vendor: Any computer dealer.
  Comment: All PC-SEAPAK documentation was prepared using WordPerfect 5.1.

Vendor Information

3COM Corporation.
Address: 3165 Kifer Road, Santa Clara, CA 95052
Tel.: (408) 562-6400 or (800) 638-3266

AST Research, Inc.
Address: 2121 Alton Avenue, Irvine, CA 92714
Tel.: (714) 863-1333

Cipher Data Products
Address: 9715 Business Park Ave., San Diego, CA 92131
Tel.: (619) 578-9100

Flagstaff Engineering, Inc.
Address: 1120 Kaibab Lane, Flagstaff, AZ 86001
Tel.: (602) 779-3341

Hewlett Packard
Address: P.O. Box 3640, Sunnyvale, CA 94088-3640
Tel.: (800) 538-8787

Matrox Electronic Systems Limited
Address: 1055 St. Regis Boulevard, Dorval, Quebec, Canada
Tel.: (514) 685-2630

Media Cybernetics
Address: 8484 Georgia Avenue, Silver Spring, MD 20910
Tel.: (301) 495-3305

MicroWay, Inc.
Address: Building #20, Cordage Park, Plymouth, MA 02360
Tel.: (508) 748-7341

Mitsubishi Electronics America, Inc.
Address: 110 New England Avenue, Piscataway, NJ 08854
Tel.: (201) 981-1414

Overland Data, Inc.
Address: 5600 Kearney Mesa, San Diego, CA 92111
Tel.: (619) 571-5555 or (800) 729-8725

Phar Lap Software, Inc.
Address: 60 Aberdeen Avenue, Cambridge, MA 02138
Tel.: (617) 661-1510

QMS, Inc.
  Address: One Magnum Pass, Mobile, AL 36618
  Tel.: (800) 523-2696

Quarterdeck Office Systems
  Address: 150 Pico Boulevard, Santa Monica, CA 90405
  Tel.: (213) 392-9851

Tektronix, Inc.
  Address: Wilsonville Industrial Park, P.O. Box 1000
  Wilsonville, OR 97070
  Tel.: (503) 682-3411

APPENDIX
HALO88 FONT STYLES

HALO88 fonts are provided along with PC-SEAPAK on a separate diskette through a licensing agreement with Media Cybernetics. The following table describing these font styles is from Media Cybernetics' manual for HALO88.

Appendix

HALO Font Styles

HALO001

HALO002

ABCDEFGHIJKLMNOPQRSTUVWXYZ1234567890
abcdefghijklmnopqrstuvwxyz!@#$%^&*()

HALO010

HALO011

HALO012

ABCDEFGHIJKLMNOPQRSTUVWXYZ1234567890
abcdefghijklmnopqrstuvwxyz!@#$%^&*()

HALO013

ABCDEFGHIJKLMNOPQRSTUVWXYZ1234567890
abcdefghijklmnopqrstuvwxyz!@#$%^&*()

HALO109

ABCDEFGHIJKLMNOPQRSTUVWXYZ1234567890
abcdefghijklmnopqrstuvwxyz!@#$%^&*()

HALO111

ABCDEFGHIJKLMNOPQRSTUVWXYZ1234567890
abcdefghijklmnopqrstuvwxyz!@#$%^&*()

HALO115

ABCDEFGHIJKLMNOPQRSTUVWXYZ1234567890
abcdefghijklmnopqrstuvwxyz!$%&*()

HALO201

ABCDEFGHIJKLMNOPQRSTUVWXYZ1234567890
abcdefghijklmnopqrstuvwxyz!$%&*()

HALO202

ABCDEFGHIJKLMNOPQRSTUVWXYZ1234567890
abcdefghijklmnopqrstuvwxyz!$%&*()

HALO203

ABCDEFGHIJKLMNOPQRSTUVWXYZ1234567890
abcdefghijklmnopqrstuvwxyz!$%&*()

HALO204

ABCDEFGHIJKLMNOPQRSTUVWXYZ1234567890
abcdefghijklmnopqrstuvwxyz!@#$%^&*()

HALO102

*ABCDEFGHIJKLMNOPQRSTUVWXYZ*1234567890
abcdefghijklmnopqrstuvwxyz!@#$%^&*()

HALO103

ABCDEFGHIJKLMNOPQRSTUVWXYZ1234567890
abcdefghijklmnopqrstuvwxyz!@#$%^&*()

HALO104

ABCDEFGHIJKLMNOPQRSTUVWXYZ1234567890
abcdefghijklmnopqrstuvwxyz!@#$%^&()*

HALO105

ABCDEFGHIJKLMNOPQRSTUVWXYZ1234567890
abcdefghijklmnopqrstuvwxyz!@#$%^&*()

HALO106

ABCDEFGHIJKLMNOPQRSTUVWXYZ1234567890
abcdefghijklmnopqrstuvwxyz!@#$%^&*()

HALO107

ΑΒΗΔΕΦΓΙΛΚΜΝΟΠΘΧΣΤΤ∝ΩΞΨΖ1234567890
αβηδεφγχικλμνοπ∇ρστυωξψζ√√→←↑↓(ϑ()

HALO108

ABCDEFGHIJKLMNOPQRSTUVWXYZ1234567890
abcdefghijklmnopqrstuvwxyz!$%&*()

HALO205

ABCDEFGHIJKLMNOPQRSTUVWXYZ1234567890
abcdefghijklmnopqrstuvwxyz!$%&*()

HALO206

ABCDEFGHIJKLMNOPQRSTUVWXYZ1234567890
abcdefghijklmnopqrstuvwxyz!$%&*()

HALO207

ABCDEFGHIJKLMNOPQRSTUVWXYZ1234567890
!$&()

HALO208

HALO250

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	January 1992	Technical Memorandum

4. TITLE AND SUBTITLE
PC-SEAPAK User's Guide
Version 4.0

5. FUNDING NUMBERS
WU-971-579-11-03

6. AUTHOR(S)
Charles R. McClain, Gary Fu, Michael Darzi, and James K. Firestone

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Laboratory for Hydrospheric Processes
NASA-Goddard Space Flight Center
Greenbelt, Maryland 20771

8. PERFORMING ORGANIZATION REPORT NUMBER
92B00032
Code 970

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
National Aeronautics and Space Administration
Washington, D.C. 20546-0001

10. SPONSORING/MONITORING AGENCY REPORT NUMBER
NASA TM-104557

11. SUPPLEMENTARY NOTES
C. R. McClain: Goddard Space Flight Center, Greenbelt, Maryland
G. Fu, M. Darzi, J. K. Firestone: General Sciences Corporation, Laurel, Maryland

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Unclassified - Unlimited
Subject Category 48

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

PC-SEAPAK is designed to provide a complete and affordable capability for processing and analysis of NOAA Advanced Very High Resolution Radiometer (AVHRR) and Nimbus-7 Coastal Zone Color Scanner (CZCS) data. Since the release of version 3.0 over a year ago, significant revisions have been made to the AVHRR and CZCS programs and to the statistical data analysis module, and a number of new programs have been added. This new version has 114 procedures listed in its menus. The package continues to emphasize user-friendliness and interactive data analysis. Additionally, because the scientific goals of the ocean color research being conducted have shifted to larger space and time scales, batch processing capabilities have been enhanced, thus allowing large quantities of data to be easily ingested and analyzed. The development of PC-SEAPAK has been paralled by two other activities that have been influential and assistive: the global CZCS processing effort at GSFC and the continued development of VAX-SEAPAK. SEAPAK incorporates the instrument calibration and supports all levels of data available from the CZCS archive.

14. SUBJECT TERMS
Ocean Color, Oceanography, Sea Surface Temperature, Image Processing, Coastal Zone Color Scanner, AVHRR, Personal Computer

15. NUMBER OF PAGES
332

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UL

National Aeronautics and
Space Administration

Washington, D.C.
20546

Official Business
Penalty for Private Use, $300

Postage and Fees Paid
National Aeronautics and
Space Administration
NASA-451

UNITED STATES POSTAL SERVICE

NASA

POSTMASTER

If Undeliverable (Section 158
Postal Manual) Do Not Return